Universitat de Lleida

Escola Politècnica Superior

Enginyeria en Informàtica de Sistemes

Treball Final de Carrera

Implementació d'un algorisme d'optimització de rutes de Picking i reposició de magatzems.

Autor: Esther Buira Adern
Director: Toni Granollers
Xavi Clua

Juny del 2010

Índex

Índex de figures

Agraïments

M' agradaria agrair la col·laboració, generositat, amabilitat i dedicació del seu temps a; Xavier Clua com a director de projecte de IFR, S.A, a Toni Granollers com a director de projecte de la Universitat i a totes aquelles persones que sempre que ho he necessitat m'han ajudat i aconsellat en tot moment. Especialment m'agradaria donar les gràcies a la Marta, a l'Antoni i al Roger ja que sense ells no hagués estat possible realitzar aquest treball.

1. Introducció

Un ERP (en anglès *Enterprise Resource Planning*) és un sistema de planificació de recursos d'una empresa que ens permet integrar informàticament tota la gestió d'aquesta. Bàsicament, es pot definir com una aplicació de gestió empresarial que integra el flux de treball de la informació i ens permet millorar els processos de les diferents àrees de l'empresa (gestió de inventari, ventes, compres, producció, gestió de projectes, recursos humans, financera, entre altres).

Els sistemes ERP integren tots els processos de negoci de l'empresa permetent gestionar i controlar diferents departaments i processos a partir d'un conjunt d'informació compartida, centralitzada i no redundant. Una altra de les característiques a destacar d'aquests sistemes són els seus dissenys modulars i adaptables als requeriments de l'empresa. En definitiva tot això ens permet millorar els processos, reduir costos i estalviar temps innecessaris.

En el mercat actual existeixen multituds de solucions ERP i mòduls de gestió (vendes, finances, producció, magatzems, etc.) cadascun dels quals té unes prestacions i uns costos diferents. Per exemple:

- Microsoft Dynamics AX
- JD Edwards
- SAP Business On
- SAP R/3

Per a la implementació d'aquest treball he escollit utilitzar el sistema ERP Microsoft Dynamics AX (a partir d'ara Axapta) ja que, d'una banda, està considerat un dels millors sistemes ERP d'Espanya i, d'altra banda, pot

arribar a tenir molta flexibilitat en les empreses, és a dir, tant pot escalar cap amunt fins arribar a donar suport a grans divisions d'empreses o escalar cap avall fins abordar els requisits més concrets de les empreses petites. A més a més, Axapta ha estat creat per adaptar-se a les diferents casuístiques de cada empresa, la qual cosa, s'aconsegueix mitjançant la configuració i la parametrització dels processos que conté d'acord amb els objectius que es proposin per a cadascun d'ells. La parametrització és un dels valors afegits fonamental que s'ha de tenir present en qualsevol sistema ERP per a adaptar-lo a les necessitats concretes de cada empresa.

Una de les característiques d' Axapta és la seva modularitat, és a dir, està dividit en diferents mòduls que es poden instal·lar en funció de les necessitats de cada client. El mòdul més important per a la realització d'aquest treball és l' anomenat Sistema de Gestió de Magatzems, també anomenat WMS (*Warehouse Management System*). Aquest mòdul és l'encarregat de controlar els processos logístics d'emmagatzemament (recepció de productes, gestió d'estocs, nivells d'inventari, ubicacions i moviments, preparació de comandes, sortides de magatzem, etc.).

Finalment, cal dir que a banda dels motius anteriorment esmentats he escollit el sistema ERP per a realitzar el treball final de carrera perquè actualment treballo a l'empresa IFR, S.A. i l'estic implantant a diferents clients. A més a més, aquest programa em permet fer un esquema de magatzem en el qual es pot definir amb tota llibertat els passadissos i les cel·les (o zones d'emmagatzemament) del magatzem . Es tracta doncs, d'una eina molt eficient en el disseny i la parametrització dels magatzems que agilitza moltíssim la realització d'aquest projecte.

1.1. Presentació del problema

Tot seguit us faré cinc cèntims de com ha estat l'evolució dels magatzems al llarg de la història. Crec que això és molt important per poder visualitzar i entendre l'objectiu principal que pretenc aconseguir en aquest projecte de final de carrera.

Antigament, el magatzem era un espai de la fàbrica on reposaven els productes. Bàsicament, eren temps que es caracteritzaven per una baixa tensió financera de l'empresa on la vida dels productes era més estable i menys volàtil, a més a més, el marge de temps per a servir els productes als clients era més ampli.

Al llarg dels anys ha estat imprescindible millorar el servei que s'oferia als clients i això ha implicat la necessitat de disminuir el termini de resposta i el nombre de carències.

A més, l' increment de la competència de les empreses ha obligat a augmentar la productivitat, cosa que, ha tingut repercussions tant en el cost industrial com en el cost logístic dels productes: inversions en magatzems, costos financers dels estocs, manipulacions, personal dedicat exclusivament al procés de Picking[1], etc.

D'altra banda, com a conseqüència de l'augment del cost logístic dels productes es planteja una nova forma de treball que afecta bàsicament a la distribució d'un magatzem, és a dir, sorgeix el concepte de gestió dels magatzems.

La gestió dels magatzems és important sobretot en aquelles empreses que es dediquen a la producció d'aliments.

[1] Procés de Picking, és el procés per a la recollida dels productes per a la realització d'una comanda d'un client. Aquest procés s'explica en capítols posteriors

En aquestes empreses els magatzems es caracteritzen per ser el lloc on es manipulen, es guarden i es conserven els productes abans de que aquest arribin als clients. Així mateix, cal realitzar un control de les existències presents en el mateix, tenint en compte la quantitat, la data de venciment d'entrega dels productes, l'adequada rotació del producte, etc.

L' emmagatzemament pròpiament dit ha de ser l'espai adequat per a la realització d'una ràpida identificació i col·locació dels productes.

És molt important que en l' emmagatzemament, el control i la manipulació dels elements que conformen un magatzem avancin al mateix ritme que la tecnologia, de manera que, puguem identificar ràpidament els productes, reduir al màxim el temps d'entrega d'aquests als clients i tot plegat això ens permeti oferir-los un millor servei. És en aquest moment on comencen a parlar de la logística del magatzem.

L'estratègia, l'estructura logística del client i els següents paràmetres: la composició de les gammes dels productes i dels articles, les propietats dels articles, les estructures de les demandes i dels proveïdors; esdevenen factors decisius per a l'elecció de la millor solució d'una ruta[2] òptima de Picking i una gestió eficient del magatzem. Definitivament, el que es vol aconseguir és una bona rendibilitat de la inversió logística a llarg termini.

Per tant, podem concloure que els principals problemes amb el quals ens trobem alhora d'establir la gestió de magatzems són els següents:

- Millorar l'eficiència de les seves operacions.

- Reduir els costos.

- Reducció dels costos beneficiant tant als clients com a l' empresa.

[2] Ruta, és el camí ha seguir per a l'usuari del magatzem per a la recollida dels productes d'una comanda de venta.

1.2. Objectius del projecte

Els objectius d'aquest projecte consisteixen en:

- L'estudi dels diferents sistemes d'emmagatzemament i distribució de magatzems logístics actuals.

- Conèixer més profundament quines són les funcions principals d'un magatzem, definir els inventaris i veure en què consisteixen.

- L'anàlisi i la implementació d'un algorisme òptim per al càlcul de rutes de Picking i de reposició en alguns dels models de magatzems estudiats aplicant paràmetres estàndards com per exemple:

- Caducitat dels productes: s'ha de tenir en compte la caducitat del producte a entregar per a segons quins clients. El transport pot variar en funció de l'adreça del client i, per tant, cal saber la data de caducitat d'aquest per a identificar el número del lot del producte que cal servir de forma que quan el rebi no hagi caducat i encara tingui un marge de dies per fer-ho.

    - Pes, volum: és important tenir en compte aquestes dues característiques en un producte alhora de distribuir-lo i també en la seva recollida. L' usuari encarregat de preparar una comanda per a un client ha de tenir present que l'objectiu principal és distribuir i recollir els productes tenint en compte el seu pes i volum i ordenar-los de més volum i més pes a menys respectivament per tal de conservar el bon estat dels productes.

- Distribució del magatzem: segons la distribució del magatzem que estudiarem la recollida de productes pot variar en funció d' on es troben ubicats. L'idoni seria que quan l'usuari acabi de preparar la comanda del client ho faci tan a prop com sigui possible del moll

de sortida del magatzem. En aquest treball estudiarem dos tipologies de magatzems, les quals les podrem veure en l' apartat 5.2.

1.3. Estructura del projecte

En el capítol 2 d'aquest treball descriurem els conceptes generals que engloben l'estructura d'un magatzem, els elements que el formen i les seves funcions principals. També explicarem que són els inventaris i quines són les parts més importants que els formen i els caracteritzen.

En el capítol 3 farem una petita introducció a Axapta i als elements que formen el sistema ERP, així com, a l'entorn de treball que s'utilitza per a la implementació de noves funcionalitats. A més a més, farem esment de la metodologia que s'utilitza i d'algunes sentències bàsiques de la programació.

En capítol 4 explicarem els diferents elements que composen el mòdul de gestió d'inventari d'Axapta, els seus mestres / formularis claus i els processos que el formen.

En el capítol 5 descriurem els requisits per a la implementació de l'algorisme i els paràmetres ha tenir en compte i les casuístiques del mateix algorisme per a la seva implementació. També podrem veure quins són els dos dissenys de magatzem que analitzarem un cop aplicat l'algorisme.

En el capítol 6 explicarem en què consisteix cadascun dels mètodes que formen part de l'algorisme i podrem veure quin és el flux de treball per a la seva realització.

En el capítol 7 mostrarem quines han estat les conclusions a les que hem arribat després d'aplicar l'algorisme de Picking i reposició en els dos dissenys de magatzem per a diferents comandes de venta amb articles que contenen característiques diferents.

Finalment, en el capítol 8 donarem a conèixer quins són els possibles treballs futurs que es poden realitzar a partir d'aquest algorisme de rutes de Picking i de reposició que s'ha implementat.

2. Conceptes genèrics per a la gestió d'un magatzem

Primerament, m'agradaria fer referència a una sèrie d'elements bàsics que ens permetran entendre millor alguns dels conceptes claus que conformen la realització d'aquest projecte.

2.1. Definició magatzems

Un magatzem es pot definir com el lloc físic on es guarden les mercaderies i en el que es desenvolupa una completa gestió dels productes que conté.

Aquesta gestió ens dóna la flexibilitat necessària per definir la distribució i disposició de les mercaderies en el nostre magatzem a mesura que canvien les nostres necessitats i garantint sempre la màxima eficàcia possible. Per aconseguir-ho és molt important establir àrees d'emmagatzemament de major i menor prioritat per tal d' optimitzar la localització dels productes, i dividir el magatzem en zones per poder-hi acomodar les diferents necessitats d'emmagatzemament que van sorgint atenent a criteris de temperatura o al ritme de rotació de l' inventari dels diferents productes. A més a més, la gestió de magatzems ens permet determinar la seqüència més lògica de les ubicacions de Picking optimitzant les rutes de Picking en funció de paràmetres com ara les zones i l'altura de les ubicacions.

També, cal destacar la logística de magatzems perquè ens ofereix una gran funcionalitat de previsió i permet una planificació a llarg termini optimitzant els nivells de l' inventari.

La planificació d'un magatzem és una eina bàsica per optimitzar la producció, la planificació de materials i la programació.

La programació de capacitat i material finit es pot realitzar simultàniament de manera que la capacitat disponible, els nivells d'inventari

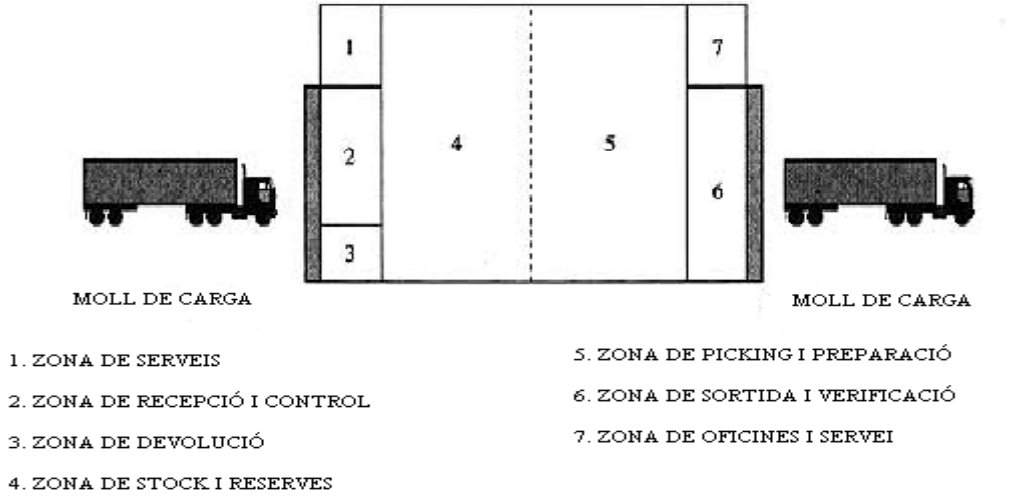
i els temps d'espera emprats en realitzar les compres i distribuir el transport es tinguin en compte alhora de planificar la seva producció. Quelcom, ens permet planificar amb més fiabilitat les ordres de compra, de producció i de transferència aconseguint optimitzar el flux de producció i les entregues dins del termini establert per als clients.

Finalment, cal destacar que avui en dia els magatzems intenten corregir els desequilibris que sorgeixen entre la oferta i la demanda i a la vegada reduir els costos.

2.2. Zones d'un magatzem

Un magatzem sol tenir unes zones que el caracteritzen. Aquestes zones generalment són comunes en els diferents tipus de magatzems existents en les diverses empreses del món laboral.

Un exemple d'un esquema de magatzem podria ser el següent [ESTC]:



Il·lustració 1 Esquema estàndard d'un magatzem

Les zones més importants d'un magatzem són les següents:

- Àrea de recepció: És la zona on es realitzen les activitats de recepció. Aquesta zona ha de tenir unes dimensions adequades al volum de mercaderies que rep i al període de temps que han de romandre en aquesta zona. El temps de permanència de les mercaderies en aquesta àrea ha de ser el mínim possible. L'àrea de recepció assumeix una de les majors activitats, ja que, és el lloc on es verifiquen els productes rebuts. Cal comprovar que la mercaderia rebuda es troba en perfecte estat i no presenta cap anomalia i verificar que l'albarà d'entrada es correspon amb la mercaderia física. Ja que, si això no es comprova i s'accepta el material rebut davant de qualsevol reclamació que es faci el proveïdor no es fa responsable de si la quantitat rebuda és la correcta o bé si el material presenta algun tipus de desperfecte.

- Àrea d'emmagatzemament, reserva o estoc: És l'espai físic dedicat a guardar els productes. Hi han dos tipus de sistemes d'emmagatzematge cadascun dels quals presenta diferents necessitats d'espai.

    - Ubicació física: Amb aquest sistema d'ubicació cada unitat d'emmagatzematge té un determinat lloc al magatzem on únicament aquest ítem pot ser emmagatzemat, de manera que, si aquesta ubicació queda buida tan sols si podrà col·locar una altra unitat de carga del mateix tipus. El nombre d'ubicacions necessàries serà l'equivalent a l'estoc màxim de cada ítem.

    - Ubicació aleatòria: Aquest sistema ens permet millorar el grau d'ubicació del magatzem. Les unitats de carga poden ser ubicades en qualsevol espai lliure, degut a que, el ritme d'entrada i sortida dels diferents productes sol ser diferent.

- Zona de Picking o preparació de comandes: Aquesta zona està destinada a la recuperació dels productes del seu lloc

d'emmagatzemament i realitzar la seva preparació per ser traslladats o transportats adequadament.

- Àrea d'entrega: És la zona on es prepara la mercaderia per a la carga i es realitza la verificació quan surt del magatzem. Un bon control a l'entrada i la sortida de les mercaderies és bàsic per a què no es produeixin diferències a l'inventari.

2.3. Funcions principals

Les principals funcions d'un magatzem són les següents:

- La recepció: És l'encarregada d'admetre les demandes esperades. Durant la recepció cal que s'efectuïn les següents operacions:

    - S'ha de disposar d'una relació de les comandes que s'han de rebre en funció del termini d'entrega.

    - Cal rebre l'enviament del proveïdor mitjançant un albarà d'entrega que ens lliurarà el transportista.

    - El receptor ha de comprovar que les dades del document d'arribada i les ordres de les comandes coincideixin .

    - Cal ordenar la descàrrega dels productes.

    - S'han de contar els articles descarregats i verificar que no presentin desperfectes. En el cas de què els articles presentin desperfectes aquests s'han de comunicar immediatament al responsable de compres.

    - S'han d'anotar els desperfectes tribats a l'albarà del transportista.

- Cal notificar l'arribada al Departament d'Etiquetatge en el cas de superfícies grans.

- La identificació: Té com a missió realitzar una inspecció general dels productes rebuts. És per això, que s'han de realitzar les següents funcions:

  - Comprovar que els productes que s'han rebut coincideixin amb els productes descrits a l'albarà d'entrega.

  - Posar els materials a disposició del Departament de Control de Qualitat per a la seva verificació.

  - Enviar les dades d'identificació al Departament de Compres i Comptabilitat amb les respectives anomalies que s'han trobat per a què aquest reclami els productes que falten, gestioni les devolucions, etc. I procedeixi a donar d'alta les entrades de material autoritzant el pagament d'aquestes.

  - Entrega del producte als responsables del transport intern perquè el col·loquin al lloc corresponent.

  - Donar d'alta els productes a l' estoc i informar on s'han ubicat.

Durant el procés de recepció i d'identificació es comprova immediatament que no hi hagin errors en les transaccions realitzades, ja que, com més aviat es detectin aquests menys perjudicis comportaran. D'altra banda, és important destacar que el registre d'entrada d'un article és l'únic document que corrobora el que s'ha rebut. Per tant, l'execució d'aquestes funcions s'han de dur a terme de la forma més acurada possible.

El Departament de Compres ha d'exigir al Departament del Magatzem que les mercaderies entregades per part dels proveïdors

siguin precisament les que han estat sol·licitades en la comanda i que no acceptin cap tipus de variació. Per aquest motiu, cal disposar de fulls de control per a la recepció de productes, els quals han de contenir la següent informació:

- Nombre de comandes.
- Quantitat entregada.
- Descripció de l'article.
- Nom del proveïdor.
- Unitat de mesura.
- Observacions.
- Nom i signatura del treballador que rep els productes.

- Custòdia i entrega: La custòdia consisteix en col·locar i guardar en el lloc adequat dins del magatzem els productes a l' espera de rebre l'ordre d'entrega. L'entrega suposa posar el material a disposició del Departament de Distribució del Material una vegada rebuda l'ordre, de forma ràpida i sense cometre errors. El magatzem en compliment de les funcions de custòdia i entrega de les mercaderies que li corresponen té les següents responsabilitats:

  - Emmagatzemar de forma segura, tècnica i físicament els productes.
  - Protegir els articles de possibles robatoris, d'una utilització no autoritzada, de desperfectes i deterioraments.

- Marcar i localitzar les mercaderies de tal forma que siguin accessibles.

- Controlar la sortida de material per tal de garantir un bon servei, a més a més, de protegir-lo contra utilitzacions no autoritzades.

- Actuar directament sobre els costos de deteriorament.

- Altres funcions més específiques

  - Mantenir les matèries primes en algun lloc segur protegides dels incendis, robatoris i possibles deterioraments.

  - Permetre a les persones autoritzades l'accés al material emmagatzemat.

  - Mantenir informat al Departament de Compres sobre les existències reals dels productes.

  - Controls en les entrades i sortides dels materials.

Finalment, cal destacar que la missió bàsica d'un magatzem es configura i es desenvolupa en el compliment de les tres funcions següents:

- Recepció d'articles. S'ha de permetre una recepció còmoda i ràpida dels productes.

- Emmagatzemament. Cal disposar d'instal·lacions adequades depenent del tipus d'articles que hi contindran i de les seves necessitats d'emmagatzemament i de manipulació.

- Entrega de productes. Es realitzarà mitjançant una sortida ràpida dels productes.

2.4. Tipologies

Els diferents tipus de magatzems que podem trobar en el món laboral es classifiquen en funció d'una sèrie de paràmetres. Depenent del paràmetre que regeixi el magatzem els podem classificar segons sigui:

- La seva relació amb el flux de producció.
- La seva ubicació.
- El material a emmagatzemar.
- La seva localització.
- La seva funció lògica.

2.5. Conceptes per a l'inventari

Podem definir un inventari com una estadística física o compteig dels materials existents per a contrastar-la amb les existències anotades en els fitxers d'existències o en el banc de dades sobre els materials. Algunes empreses li donen el nom d'inventari físic perquè es tracta d'una estadística física o palpable d'aquells productes que hi ha en les existències de l' empresa.

L' inventari físic s'efectua periòdicament, casi sempre, coincideix amb el tancament del període fiscal de l'empresa a efecte del balanç contable. En aquesta ocasió l'inventari es fa per a tota l'empresa.

D'altra banda, els inventaris representen una proporció significativa dels actius en la majoria d'empreses. És per aquest motiu, que els processos administratius que donin com a resultat la minimització del percentatge

d'inventari total poden presentar grans estalvis de diners per a l'empresa. Els inventaris representen bens destinats a la venta en curs dels negocis.

Amb els inventaris l'empresa pot realitzar les seves tasques de producció i de compra economitzant els recursos i també atendre als seus clients amb més rapidesa, optimitzant totes les activitats de l'empresa. No obstant això, hi ha un inconvenient que és el cost de manteniment.

L'objectiu principal del control d'inventaris és aconseguir que l'empresa disposi de la quantitat apropiada de matèria prima, d'altres materials i dels productes acabats en el lloc adequat amb el temps oportú i amb el menor cost possible, és a dir, poder arribar a tenir una bona aproximació del consum real d'aquesta a un baix cost. Els costos excessius en els inventaris poden ser conseqüència de les males decisions en l'establiment d'un sistema. És molt important doncs, tenir un bon sistema d'emmagatzemament i control d'inventari per a la reducció d'aquests.

Cal tenir en compte que l'eficiència del control d'inventaris pot afectar la flexibilitat d'operació de l'empresa. Per exemple: dos empreses pràcticament idèntiques, amb la mateixa capacitat d'inventari, però amb grans diferències de flexibilitat en les seves operacions poden tenir inventaris desbalancejats. Això es degut, a controls ineficients dels graus de flexibilitat els quals ocasionen que en algun moment hi hagi abundància d'alguna matèria prima i escassetat d'una altra. Per tant, podem dir que a menor eficiència en el sistema de control d'inventaris major serà la necessitat d'inversió. Com a conseqüència d'això, les altes inversions d'inventari tindran un impacte desfavorable en la utilització de l'empresa.

Per concloure, podem dir que un dels principis bàsics del control intern d'un magatzem és garantir l'exactitud entre les existències físiques dels productes emmagatzemats i els registres corresponents dels mateixos en el sistema informàtic. Per facilitar el recompte cada referència ha d'estar ubicada en un únic lloc del magatzem. Cal dir, que les diferències d'inventari son inevitables i poden ser degudes a múltiples causes que van

des d'un simple robatori del producte, a equivocacions del Picking, fins a errors més complicats de resoldre. Per tant, és molt important la realització periòdica de recomptes físics dels productes i comparar-ne el contingut amb els registres d' estocs.

2.5.1. Criteris de valoració d'inventari

El sistema de gestió d'un magatzem és el sistema que determina els criteris per seleccionar el material que ha de sortir del magatzem per atendre una petició concreta. La importància d'aquest sistema radica en què incideix directament sobre el període de permanència dels productes al magatzem.

Per configurar el sistema de gestió de magatzems dins d' Axapta disposem del formulari Grups de Models d'Inventari. Aquest ens permet crear i mantenir grups de models d' inventari. Aquests grups contenen valors que determinen com es controlen i es manipulen els productes quan es reben, s'emeten i també permeten determinar el càlcul del consum de l'inventari.

Pel que fa als sistemes de gestió de magatzems hi han 6 criteris de valoració d'inventaris:

- LIFO (última entrada, primera sortida) : És un model d'inventari en el qual les últimes recepcions s'emeten abans. Les emissions de l'inventari es liquiden en les últimes recepcions segons la data de la transacció d'aquest. Si s'utilitza LIFO podem escollir l'opció de marcar les transaccions de l'inventari, de manera que, un article específic es liquidi en una emissió concreta en lloc d'utilitzar la regla LIFO. És el millor criteri en èpoques d'inflació perquè el cost del producte quedarà valorat a preus actuals. Les existències finals quedaran infravalorades (valorades a preus antics). Es recomana realitzar el tancament del inventari de forma periòdica.

- Data LIFO: És un model d'inventari basat en el principi LIFO on les emissions de l'inventari es liquiden enfront a les últimes recepcions rebudes i en funció de la data de la transacció de l'inventari. Amb la data LIFO si no hi ha cap recepció abans de l'emissió aquesta es liquida front a qualsevol recepció que aparegui després de la data de l'emissió. Varies emissions en una mateixa data es poden liquidar en l'ordre de l' última emissió, última recepció.

- FIFO (primera entrada, primera sortida): És un model d'inventari en el qual les primeres recepcions adquirides s'emeten primer. Les emissions d'inventari es liquiden davant les primeres recepcions en el mateix en funció de la data física de la transacció de l'inventari. En èpoques d'inflació presenta un problema i és que el cost del producte queda infravalorat (valor per sota del seu valor real). Les existències finals queden valorades a preus actuals. Es recomana realitzar el tancament de l'inventari de forma periòdica.

- PMP (preu mig ponderat): Valora les sortides del magatzem amb la mitja ponderada de totes les entrades.

- Preu cost mig a data: La data mitja ponderada és un model d'inventari basat en el principi de la mitja ponderada, segons la qual, les emissions de l'inventari es valoren segons el valor mig dels articles que es reben en l' inventari diari i diferent en el període de tancament. Al realitzar el tancament de l'inventari, totes les recepcions d'un dia es liquiden front una emissió virtual que retè la quantitat rebuda total i el valor per al dia corresponent.

És important mantenir el mateix criteri de valoració per a tots els inventaris de l'empresa.

Axapta solsament utilitza quatre d'aquests criteris, els quals són LIFO, FIFO, PMP i preu mig a data.

2.5.2. Tipologies d'inventari

Podem classificar els inventaris en funció de [TIPO]:

- Les característiques de l' empresa:

    - Inventari de mercaderies: El constitueixen tots aquells bens que pertanyen a la empresa tant si són comercials com mercantils. Aquests bens es compren per a tornar-los a vendre sense ser modificats.

    - Inventari de productes acabats: Són tots aquells bens adquirits per l'empresa manufacturera o industrial que són transformats i després venuts com a productes elaborats.

    - Inventari de productes en procés de fabricació: L'integren tots aquells bens adquirits per les empreses manufactureres o industrials que es troben en procés de manufactura. La seva quantificació es fa d'acord a la quantitat de materials, mà d'obra i costos de fabricació aplicables a la data de tancament de l'inventari.

    - Inventari de matèries primes: El formen tots aquells materials amb els quals s'han d'elaborar els productes però que encara no s'han rebut.

    - Inventaris de subministres de fàbrica: Són els materials amb els que s'elaboren els productes però que no poden ser quantificats d'una manera exacta (pintura, claus, lubricants, etc.)

- Quan s'efectuen:

    - Anual : Sols s'efectua un cop a l'any.

- Inconvenients**:**
    - Paralització de l'activitat de l'empresa.
    - Alt nivell de diferències entre existències físiques dels productes emmagatzemats i els registres del sistema administratiu.
    - Dificultat per seguir el moviment històric de les diferències.

- Mensual: S'efectua un cop al mes.

- Rotatiu: Control del productes diari per zones, ubicacions, consum o valor.

    - Avantatges:
    - Detectar i modificar les diferències existents dins de l'operativa del mateix mes.

- Aleatori: Control a l'atzar de diferents referències de productes.

    - Avantatges:
    - Ajustar els aprovisionaments i les ventes dels productes emmagatzemats.
    - Facilitat en el seguiment de l'històric.

- Dinàmic: Control dels productes a la mateixa vegada que s'efectua la seva ubicació.

- Avantatges:
    - Quadre diari dels productes.
    - Moviment històric totalment actualitzat.

2.5.3. Dimensions d'inventari

Les dimensions d'inventari constitueixen una eina molt important per classificar l'inventari en funció de l'emmagatzemament i de les característiques dels articles, de forma que, es pugui disposar d'una vista detallada de l'inventari sempre que es necessiti.

En Axapta hi han dos grans grups de dimensions:

- Dimensions d'emmagatzematge:
    - Identificació magatzem.
    - Identificació ubicació.
    - Identificació pallets.
    - Números de sèrie.
    - Números de lot.

Utilitzarem el nombre de lot i el nombre de sèrie per realitzar el seguiment dels productes dins del nostre inventari.

A més a més, cal dir que una de les dimensions més importants de l'inventari és la ubicació. La ubicació ens dóna la informació necessària per localitzar el producte de forma ràpida al magatzem indicant-nos el passadís i el prestatge on es troba.

- Dimensions d'articles:
    - Configuració.
    - Tamany.
    - Color .

Aquestes tres dimensions poden ser modificables, podem canviar-los-hi el nom i en lloc d'utilitzar el color poder utilitzar com a dimensió, per exemple: la resistència, el pes, etc. d'un producte.

2.5.4. Procés Picking

En el camp de la logística s'anomena Picking al procés de recollida de material extraient unitats o conjunts empaquetats d'una unitat d'empaquetament superior que conté més unitats que les que s'han extret.

Podem tenir un Picking d'unitats quan s'extreuen productes unitaris d'una caixa o, Picking de caixes, quan es recullen caixes d'una paleta o d'un contenidor.

El Picking és un procés bàsic en la preparació de comandes dels magatzems. Afecta en gran mesura a la productivitat de tota la cadena logística. Normalment és un procés intens en mà d'obra i la seva optimització i mecanització una de les formes de millorar el rendiment de la cadena de logística interna de les empreses [PICK].

La seva millora consisteix, com és lògic, en eliminar les parts improductives del procés. La part més improductiva sol estar relacionada amb el desplaçament entre les diferents ubicacions on s'ha d'efectuar el procés de Picking dels diferents productes. Per eliminar aquets desplaçaments tenim dos posicions normalment divergents. La primera, consisteix en la modificació dels procediments per disminuir els desplaçaments, bàsicament, agrupant les comandes i, la segona, en

optimitzar amb maquinària el transport dels contenidors de productes sobre els quals es realitzarà el procés de Picking. La primera de les opcions esmentades basa la seva eficàcia en el concepte estadístic de ruta òptima, una ruta òptima és aquella que recorre de forma òptima les diferents posicions de recollida del producte i comença a tenir rellevància a partir d'un cert número de línies de la comanda a servir en el lot, per la qual cosa, la seva eficàcia és directament proporcional al número de comandes agrupades per al seu servei simultani.

Podem concloure doncs, que el procés de Picking és sens dubte l'activitat d'un magatzem que té el valor afegit més alt. Això el fa un dels processos claus del magatzem logístic particularment en el moment d'un procés de reducció de costos.

Per a la preparació de comandes és necessari disposar d'un document, llista de Picking o preparació de comandes, el qual ha d'informar de:

- Les referències, la descripció, les quantitats i les ubicacions del magatzem de les referències que s'han de preparar.

- Les instruccions específiques de preparació: empaquetament, disposició dels producte en el paquet, etc.

2.5.5. Procés Reposició

El procés de reposició és aquell procés que controla que les ubicacions destinades per al Picking no es quedin sense productes, de tal forma, que si en una ubicació de Picking s'estan acabant els productes l'omplirem amb els productes que tenim a les ubicacions destinades a la reposició de material. Normalment les ubicacions de Picking i de reposició es troben conjuntes en la mateixa prestatgeria, per exemple: podríem tenir una ubicació de Picking damunt d'una ubicació de reposició o a l'inrevés.

2.6. Objectius dels magatzems

El paper dels magatzems en la cadena de subministrament ha anat evolucionat al llarg dels anys i, han passat de ser unes simples instal·lacions dedicades a l'emmagatzemament a convertir-se en centres enfocats al servei.

Actualment, el disseny dels magatzems tenen per objectiu facilitar la rapidesa de la preparació de les comandes, la precisió dels mateixos i la col·locació més eficient de les existències, per tal d'aconseguir, cicles de preparació de comanda més ràpids millorant el servei al client.

Aquest projecte sorgeix doncs de la necessitat de reduir el temps de servei al client, de manera que, es minimitzi el temps de manipulació de les operacions de Picking i de reposició. D'altra banda, analitzar la distribució dels diferents dissenys de magatzems estudiats, fent ús d'una bona distribució que ens permeti trobar la ruta més òptima per a la realització de la demanda que hem d'oferir-li al client i, finalment, escollir dels diferents dissenys de magatzems estudiats el més eficient per a l'aplicació de l'algoritme.

És per aquest motiu, que es planteja la possibilitat de crear un algorisme que minimitzi el temps de preparació de la comanda i redueixi el temps de Picking i reposició de les matèries. Els principals objectius que s'intenten aconseguir en la implementació d'aquest algorisme són:

- Reduir els temps de demora de l'atenció al client.

- Aconseguir la millor redistribució de la zona d'emmagatzemament tenint en compte una sèrie de paràmetres. Per tant, abans de la realització de l'algorisme ens cal estudiar els diferents paràmetres a tenir en compte per a una bona organització, els quals són:

- Reduir les distàncies recorregudes en la manipulació dels productes del magatzem.

- Millorar el control de l'inventari, a fi que, ens permeti controlar i gestionar de manera eficaç dins de l'organització i que ens asseguri un flux perfecte de la informació i dels productes.

- Finalment, de l'estudi realitzat mitjançant la implementació de l'algorisme dels diferents dissenys de magatzem existents, concloure, quin és el millor i perquè. És a dir, quin disseny de magatzem presenta l'estructura ideal i més òptima per aplicar l'algorisme .

3. Introducció a Axapta

En concret Axapta i els ERP en general són solucions pensades per arribar a oferir una gran funcionalitat. Les solucions de gestió d'un ERP han d'englobar qualsevol requeriment que es pugui presentar en una empresa sigui quina sigui l'activitat a la que aquesta es dediqui.

En aquest capítol, explicarem bàsicament la funcionalitat que ofereix l'eina escollida per realitzar aquest projecte des del punt de vista de la funcionalitat estàndard que ofereix l'aplicació, a més a més, podrem veure la facilitat que ofereix aquest sistema per desenvolupar nous processos. D'altra banda, revisarem l'entorn de desenvolupament d'Axapta per a la creació de nous processos, anomenat Entorn de MorphX, així com el seu llenguatge de programació (X++).

3.1. Entorn MorphX

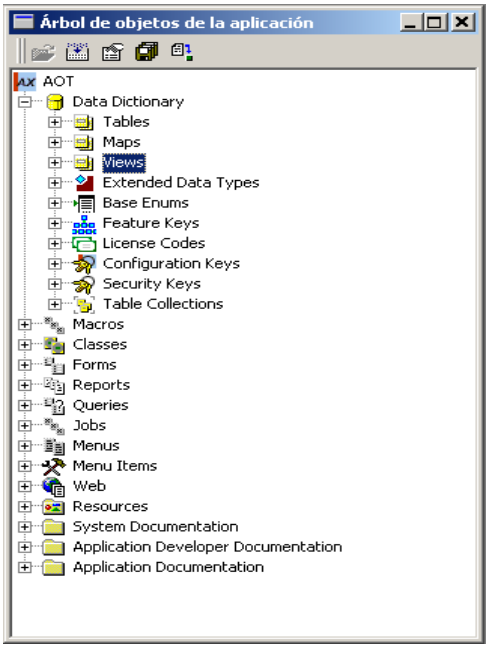
3.1.1. Definició

L'entorn de desenvolupament d'Axapta s'anomena MorphX. Es pot considerar com un entorn integrat de desenvolupaments (*Integrated Development Enviroment* o IDE), ja que, engloba les funcionalitats de disseny, edició, compilació i depuració. Està dissenyat per facilitar la modificació de la funcionalitat d'Axapta.

A més a més, també es pot caracteritzar com un entorn de desenvolupament integrat de Microsoft Dynamics AX que permet a programadors i usuaris dissenyar gràficament tipus de dades enumerades, taules, consultes, formularis, menús i realització d'informes. MorphX suporta la tecnologia "*drag and drop*[3]" i és molt intuïtiu la qual cosa facilita als usuaris finals crear els seus propis informes i consultes [Entorn MorphX].

[3] De l'Inglés 'Drag and Drop', arrastrar i soltar, és una expressió informàtica que fa referència a l'acció de moure amb el ratolí objectes d'una finestra a una altra o entre les mateixes parts d'una finestra. [Drag and Drpo]

La part central de MorphX és una estructura Gràfica en arbre que conté tots els objectes de l'aplicació. Aquesta part central s'anomena AOT (*Application Object Tree*). Podem dir doncs, que tot el que fa referència a Axapta o podem trobar en l'AOT.



Il·lustració 2 Arbre d'objectes de l'aplicació

Com podem observar en la figura, l'AOT presenta una estructura en forma d'arbre similar a la que presenta l'Explorador de Windows.

3.1.2. Conceptes importants de MorphX

Algunes de les terminologies i mecanismes central orientats a objectes apareixen quan desenvolupem amb MorphX. A continuació es farà una breu explicació dels conceptes més importants.

- Classes de sistema: una classe de sistema (*system class*) és una interface[4] de funcionalitat definida a MorphX, per exemple: per crear o executar un formulari.

[4] Interface, part del sistema (majoritariament gràfic) que permet la interacció o comunicació de la persona (usuari) amb l'ordinador.

- Classe: una classe (*class*) defineix les interfícies d'un objecte. Ensenya o explica com construir un objecte d'un tipus particular. Una característica essencial d'una classe és que podem crear noves instàncies (objectes) de la classe.

  Els formularis, informes i inclús les taules són exemples de classes: MorphX conté una definició de classe que defineix el que succeeix exactament quan un objecte de cada tipus és creat.

- Controls: un control és un objecte gràfic, com una caixa de text (*text box*), una casella de verificació (*check box*) o un botó de comando (*command button*) que podem situar en un formulari o en un informe quan el dissenyem per a què mostri la informació requerida, realitzi una acció o aconseguir que el formulari o l'informe sigui més senzill de llegir o interpretar.

- Origen de dades: un origen de dades (*Data Source*) conté les variables de dades que utilitzen un formulari o consulta. Aquestes variables de dades poden ser una o més taules o camps individuals de les taules.

- Dissenys: un node de disseny (*Designs*) proporciona accés a l'usuari per definir l'aspecte d'un formulari o d'un informe.

- Encapsulació: l'encapsulació significa que les dades en el sistema són definides dins dels mètodes i sols poden ser modificades des dels propis mètodes.

- Final: final és un modificador de la classe o d'un objecte que indica que la classe o mètode no pot ser ampliat o sobrecarregat.

- Herència: l'herència és un concepte fonamental de MorphX. Significa que el que és definit en nivells inferiors del sistema són

automàticament accessibles, o el que es el mateix, hereten de nivells superiors.

- Objectes: els objectes són el concepte central de MorphX. Qualsevol formulari i qualsevol control és un objecte. La base de dades també és un objecte. En definitiva, qualsevol cosa present en el sistema és un objecte.

  Els objectes són creats a partir de classes. Decidim per tant que un objecte és una instància a una classe.

  Els objectes proporcionen una forma lògica convincent d'organització de les dades i dels procediments. Els objectes estan encapsulats, la qual cosa, significa que contenen tant el seu codi com les seves dades.

  Per utilitzar un objecte hem de mantenir una referència amb ell mitjançant una variable del mateix tipus que l'objecte al que volem fer referència.

- Mètodes: els mètodes són tasques que podem fer que un objecte les pugui realitzar.

- Propietat: les propietats són dades que descriuen un objecte. Cada tipus d'objecte té diferents tipus de propietats. Un mètode típicament té poques propietats, una de les quals, per exemple: defineix on volem que s'executi. Per altra banda, un control té aproximadament unes 50 propietats que defineixen el color, la posició, etc.

- Consulta: una consulta és un mecanisme de filtratge que permet recuperar les dades que ens interessen a partir de les taules de la base de dades. Les consultes són utilitzades normalment com l'origen de dades en els formularis i informes.

3.1.3. Elements de l' Application Object Tree

A l' obrir l'AOT s'observen únicament els nodes principals. De cadascun d'aquests nodes "pengen" nodes secundaris o bé els anomenats objectes de l'aplicació. A continuació descriurem breument el contingut de cadascun d'aquests nodes:

- Estructura de Dades o "*Data Dictionary*": és el punt d'accés a la configuració de la base de dades, donat que des d'aquí podem crear taules, camps enumerats, modificar les propietats de qualsevol element que conté, etc.

- Macros: una macro conté línies de codi que poden resultar útils en varis punts de l'aplicació. El seu objectiu és crear sentències reutilitzables d'una forma senzilla i amb un únic punt de manteniment.

  Una macro no pot ser executada de forma independent sinó que s'ha d'utilitzar en qualsevol part del sistema en la que es pot introduir codi.

  MorphX disposa d'un preprocessador de macros que substitueix la macro per línies de codi corresponents. Aquesta substitució és realitza en temps de compilació

  Donat que el llenguatge X++ no contempla la utilització de constants, les macros s'utilitzen també per a la definició de constants.

- Classes: les classes implementen els processos que poden ser executables. Es tracta d'objectes globals que poden ser utilitzats en qualsevol part de l'aplicació.

  Cada classe està composada per una declaració de la classe (*ClassDeclaration*) que conté la definició de propietats de la classe i, una sèrie de mètodes que actuen sobre les variables donades.

- Formularis: composen la interfase de l'usuari amb la base de dades.

- Informes: els informes permeten a l'usuari presentar la informació de la base de dades que vol veure i presentar-la en un determinat ordre.

- Consultes: són filtres que permeten extreure la informació de la base de dades. Es poden utilitzar com a font de dades en els formularis o bé en informes o codis.

- Treballs: caracteritzats per petits trossos de codi que ens serveixen per provar alguna funcionalitat implementada de forma individual llençada des del mateix AOT.

- Menús: L'usuari final accedeix a la funcionalitat de l'aplicació mitjançant menús. Els menús permeten al programador organitzar desenvolupaments de manera que l'usuari pot accedir a les mateixes de forma senzilla i lògica. No obstant això, l'usuari no accedeix directament als objectes de l' aplicació a través dels menús sinó que ho fa mitjançant els menús *ítems*.

- Menús *Ítems*: nivell intermedi entre la interfase i els objectes de l'aplicació. Existeixen tres tipologies de menús *ítems*:

    - *Display*: per a formularis.

    - *OutPut*: per a informes o menús.

    - *Actions*: per a consultes o classes.

- Web: és el lloc del AOT on tenim tots els objectes propis pel portal integrat d'Axapta.

- Recursos: és el node que agrupa recursos que utilitzarà Axapta, com per exemple: imatges.

- Documentació del sistema: és on tenim la relació complerta de tots els objectes del sistema.

- Documentació de l'aplicació: conté tot el sistema d'ajuda per als objectes de l'aplicació.

3.1.4. El menú contextual

Des de qualsevol node polsant el botó dret del ratolí es desplega un menú que presenta una sèrie d'eines. Les opcions del menú seran diferents en funció del node en el qual ens trobem. En aquest apartat descriurem algunes d'aquestes opcions:

- Obrir finestra nova: obre una nova finestra amb el node seleccionat i els seus subnodes. A la finestra nova el node seleccionat es presenta com el node principal.

- Obrir: activa l'objecte seleccionat, és a dir, obre formularis. Llença l'execució d'informes i de classes, etc.

- Nou...: crea un nou objecte que s'insereix en el node seleccionat.

- Buscar: ens permet buscar objectes en l'AOT. El que es presenta a l'usuari és similar al de cerca d'arxius o carpetes de Windows.

- Exportar: genera un arxiu de text amb la informació sobre el node seleccionat i els seus elements.

- Actualitzar (*Refresh*): actualitza els objectes de l'aplicació, de manera que, els canvis realitzats per un usuari siguin accessibles a la resta d'usuaris de l'aplicació. Aquesta funcionalitat té sentit en entorns en els que variïs usuaris o clients tenen accés a les mateixes dades.

- Restaurar (*Restore*): torna a llegir la informació de l'objecte des del disc, de manera que, es descarten els canvis realitzats des de l'última vegada que l'objecte es va guardar.

- Bloquejar (*Lock*): bloqueja el node seleccionat, per tal que, la resta d'usuaris que també treballen en la mateixa aplicació no puguin modificar-lo.

- Desbloquejar (*Unlock*): desbloqueja un node bloquejat prèviament per l'usuari.

- Capes (*Layers*): mostra l'objecte en totes les seves capes.

- Add-Ins: presenta un submenú que dóna accés a una sèrie d'eines de MorphX.

- Sincronitzar: sincronitza el diccionari de dades amb la base de dades SQL que és la que utilitza Axapta per guardar la informació.

Sempre que es realitzin canvis globals en el diccionari de dades s'ha de sincronitzar la base de dades SQL. S'entén per canvis globals aquells que no són específics a una taula. Donat que la sincronització pot afectar a un gran número de dades i, per tant; requereix molt de temps. Aquest procés s'executa de forma automàtica.

3.2. Llenguatge de l'entorn MorphX

3.2.1. Introducció

El llenguatge que està per darrera del MorphX s'anomena X++ on la seva sintaxis és molt semblant a llenguatges com el C++ o el JAVA. Ja que, MorphX no ens permet dur a terme qualsevol activitat, X++ ens ofereix la possibilitat de realitzar qualsevol modificació o requeriment que sorgeixi en tot moment.

X++ és un llenguatge senzill i fàcil d'aprendre que permet que la majoria de programadors l'utilitzin sense massa dificultats.

És un llenguatge orientat a objectes, cosa que, ens permet beneficiar-nos dels avantatges de les metodologies modernes de desenvolupament del software que s'acobla perfectament en aplicacions servidor/client.

X++ ha estat dissenyat per a objectes, a més a més, proporciona una plataforma de desenvolupament orientada a objectes neta i eficient.

El llenguatge X++ utilitza principis de la programació orientada a objectes com l'encapsulació, l'herència, les classes, els objectes, els mètodes i les propietats.

Malgrat que C++ i Java van ser rebutjats com a llenguatges per utilitzar MorphX, l'aspecte de X++ és molt similar al d'aquests dos llenguatges, tot i que, s'ha eliminat la complexitat innecessària que presentaven aquests. A més a més, com que MorphX és una plataforma per construir sistemes complexes de gestió empresarial i de comptabilitat. El llenguatge X++ també inclou un gran número d'instruccions comuns de SQL com a part integrada del llenguatge. El fet que, X++ tingui una sintaxi similar a llenguatges ja existents com els comentats anteriorment fa que sigui un llenguatge familiar per a la majoria d'implantadors del software.

3.2.2. Característiques

Veurem les característiques del llenguatge en 4 grans grups:

- Llenguatge senzill i orientat a objectes: aquest llenguatge es caracteritza per ser un llenguatge senzill i perquè pot ser utilitzat ràpidament si es coneix la metodologia de la programació orientada a objectes. Els conceptes fonamentals de X++ són assimilats ràpidament fent que els programadors siguin productius des d'un bon començament.

- Llenguatge robust: el llenguatge X++ ha estat dissenyat per crear un software molt fiable. Proporciona comprovacions molt amplies en temps de compilació, seguides d'un segon nivell de comprovacions en temps d'execució.

- Llenguatge d'alt rendiment: és molt important que sempre es tingui en compte el rendiment a l'hora de treballar o de realitzar qualsevol tipus de modificació en el sistema. El llenguatge X++ aconsegueix un alt rendiment adoptant un esquema de treball, en el qual, l'intèrpret pot treballar a la màxima velocitat sense necessitat de comprovar l'entorn en el temps d'execució.

- Llenguatge interpretat i dinàmic: en una plataforma interpretada com el llenguatge X++, la fase d'enllaç d'un programa és senzilla, incremental i lleugera. D'aquesta forma, ens beneficiem d'uns cicles de desenvolupaments molt més ràpids comparats amb els cicles de compilació i proves tradicionals.

3.2.3. Metodologia de la programació

3.2.3.1. Introducció

Tot objecte pot identificar-se per l'estat en el que es troba i pel seu comportament. A efectes de la programació l'estat d'un objecte es defineix mitjançant variables, mentrestant que, el seu comportament està definit per mètodes. Els mètodes actuen sobre les variables de l'objecte, d'aquesta manera aconsegueixen evolucionar el seu estat.

Les variables d'un objecte sols són accessibles directament per mètodes propis de l'objecte mai des de l'exterior.

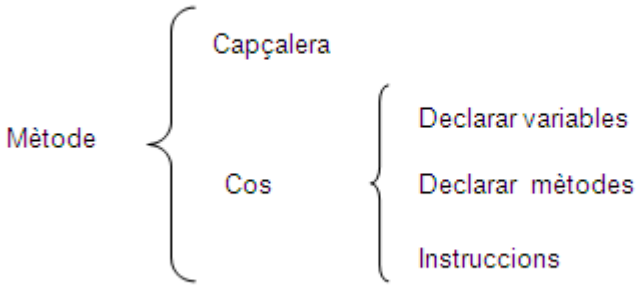
En Axapta podem trobar mètodes en tots els objectes que composen l'aplicació, és a dir, en:

- Taules
- Formularis
- Informes
- Consultes
- Classes

En cadascun d'aquests objectes els mètodes es creen de la mateixa manera que la resta d'elements de l'aplicació, o sigui, utilitzant l'Arbre d'objectes de l'Aplicació.

3.2.3.2. Estructura dels mètodes

Un mètode està format per una capçalera i un cos. Aquest a la vegada està composat per la declaració de variables i altres instruccions. L'estructura dels mètodes seria la següent:



Il·lustració 3 Estructura dels mètodes

3.2.3.2.1. Capçalera

La capçalera d'un mètode té el següent aspecte (entre [] es mostren els valors opcionals):

[Modificador] TipoDadaRetorn NomMètode ([Llistat de paràmetres])

El tipus de dada de retorn pot contenir els següents valors:

- Tipus de dades: enter, real, caràcters (*strings*) ...
- Void: s'utilitza quan el mètode no retorna res.
- Qualsevol tipus (*anytype*): significa que el mètode pot retornar qualsevol tipus de dades.

Sempre que un mètode retorna un valor s'ha d'especificar el tipus del valor que retorna i incloure-hi una instrucció de retorn (*return*...).

3.2.3.2.2. Cos

El cos d'un mètode tindrà el següent aspecte:

{
[Declaració de variables]
[Declaració de mètodes]
[;]
Instruccions;
}

La declaració de variables i mètodes es descriu a continuació.

- Declaració de variables: una variable és un punter a una posició de memòria on s'emmagatzema informació d'un tipus de dades específic.

  Totes les variables han d'estar declarades abans de poder ser utilitzades. S'ha de remarcar que X++ no permet que la declaració de variables es barregi amb les instruccions del llenguatge. En altres paraules, el llenguatge X++ requereix que les variables siguin declarades abans de qualsevol instrucció. La sintaxi de la declaració de variables és la mateixa tant si és una variable simple o una variable d'objecte. En Axapta hi han tres formes de declarar variables:

  - Declaració simple: la declaració simple és la més utilitzada, ja que, és ràpida i la majoria de variables són simples. La sintaxi seria la següent:

    TipusDada IdentificadorVariable

    El TipusDada és qualsevol tipus de dada del llenguatge X++. El IdentificadorVariable és bàsicament un nom on el primer

caràcter ha de ser una lletra que pugui estar seguida per lletres o números.

A continuació es mostren alguns exemples de declaració de variables:

int i; //Declaració d'una variables tipus enter (numèrica)

CustAccount custAccount; //Declaració d'una varibale tipus taula (en aquest cas, declarem la taula de Clients, anomenada CustAccount en Axapta)

- Declaració amb inicialització: De vegades, ens interessa que una determinada variable tingui un valor diferent al valor per defecte en el mateix moment en el que es crea la variable. El llenguatge X++ permet la inicialització de variables en la sentència de declaració.

  La inicialització es porta a fi afegint una sentència d'assignació a la declaració de la variable.

  Un exemple d'aquest tipus de declaració podria ser el següent:

  Real pi = 3.14159265359;

  Existeix una altra sintaxis per inicialitzar l'objecte, ja que, aquets són inicialitzats invocant el mètode de la classe. Un exemple d'inicialització seria el següent:

  Class classObject = newClass();

La qual cosa, declararia un objecte anomenat *classObject* de la classe *Class* i l'inicialitzaria.

- Declaració múltiple: de vegades necessitem més d'una variable del mateix tipus. En aquests casos, pot ser una pèrdua de temps haver d'escriure tota l'estona el tipus de dada davant de la variable que volem declarar. És per aquest motiu que, X++ ens permet declarar més d'una variable a la mateixa sentència de declaració. La sintaxis seria la següent:

  int i, j;

  Un exemple de declaració múltiple seria el següent:

  Real a, b = 1.55;

En aquesta sentència es declaren dos variables del tipus real anomenades a i b, inicialitzades amb el valor 1.55.

Dins d'un mateix mètode podem fer referència a l'objecte al qual pertany mitjançant la variable *this*. Aquesta variable ens dóna accés a tots els mètodes i propietats del mètode.

- Declaració mètodes: els problemes complexes del món real solen ser més fàcils de resoldre si els dividim en problemes més petits que puguin ser resolts de forma independent els uns dels altres. El llenguatge X++, ens permet introduir mètodes dins d'altres mètodes.

3.2.4. Classes

Les classes és un model de software que té una identitat pròpia amb una sèrie de propietats i mètodes.

Les propietats són les dades per a la classe i els mètodes són la seqüència de sentències que operen amb les dades. Normalment les propietats són pròpies d'un objecte, és a dir, tots els objectes són construïts a partir d'una definició de la classe i tindran la seva pròpia copia de propietats. Aquests objectes diferents són coneguts com instàncies.

Una classe no és un objecte. Una classe pot ser considerada com un avantprojecte que defineix com podrà comportar-se un objecte quan sigui creat des de les especificacions dictades per la classe. Podem obtenir objectes concrets per instanciar una classe definida prèviament. Així com, podem instanciar molts objectes d'una mateixa classe.

A continuació podem veure una declaració bàsica d'una classe molt simple anomenada Point:

```
Class Point
{
        double x;     //instància d'una propietat
        double y;     // instància d'una propietat
}
```

Aquesta declaració simplement defineix una plantilla de com objectes tipus Point poden ser instanciats.

3.2.4.1. Objectes

Gairebé, tot pot ser considerat com un objecte. Si posem un cas pràctic ens serà molt més senzill realitzar l'explicació. Per exemple: el d'un cotxe amb control remot. El cotxe consisteix amb funcionalitat i dades. Les dades estan influïdes pels mètodes o la funcionalitat velocitat, direcció, rotació, i així successivament, són dades o variables. Accedim als mètodes quan utilitzem el control remot. Per exemple: enviem un missatge perquè el cotxe es mogui cap endavant, sabem que és un mètode (funcionalitat) que pertany a l'objecte cotxe. El missatge és interpretat pel cotxe que activa el mètode de moure's cap endavant, el qual, resulta ser un moviment de rodes i un moviment cap endavant. El control remot és el manipulador que s'utilitza per accedir a l'objecte o per apuntar a l'objecte. El codi seria semblant a:

ClassName Myhandle=new ClassName();

Els objectes interactuen amb altres objectes de l'aplicació. Un objecte interactua amb altres objectes enviant un missatge que invoca una operació en l'objecte que el rep. L'objecte que rep el missatge sap com reaccionar davant d'aquest missatge rebut, així com, d'altres objectes saben com reaccionar al mateix missatge, però; pot ser que no tots els objectes reaccionin de la mateixa manera.

3.2.4.2. Exemples d'alguns mètodes que poden contenir les classes

3.2.4.2.1. ClassDeclaration

En aquest mètode és on podem escriure la declaració de les variables. Per defecte aquest mètode està buit. També és on se li assigna un nom a la classe.

Exemple:

```
class CustOverdueExpense
{
        CustPaymExpense         custPaymExpense;
        LedgerAccount           ledgerAccount;
        LedgerJournalTrans      ledgerJournalTrans;
}
```

3.2.4.2.2. New

Aquest constructor cridat automàticament quan l'objecte és creat per l'operador new. Sols ser utilitzat per inicialitzar les propietats en l'objecte nou. A continuació veurem un exemple de la utilització del mètode new en la classe Point:

```
Void new(double a=10, double b=10)
{//Constructor que inicialitza a un valor per defecte
        x = a;
```

```
            y = b;
        }
```

3.2.4.2.3. Finalize

Si es vol acabar un objecte alliberant l'espai que ocupava utilitzarem el mètode *Finalize*. Tot i que el mètode *Finalize* no conté cap mena de codi, l'execució del mètode té un efecte important. Quan es crida a aquest mètode, MorphX tanca l'objecte, la qual cosa, significa que utilitzant l'objecte es pot treure a ell mateix de la memòria.

3.2.4.2.4. Main

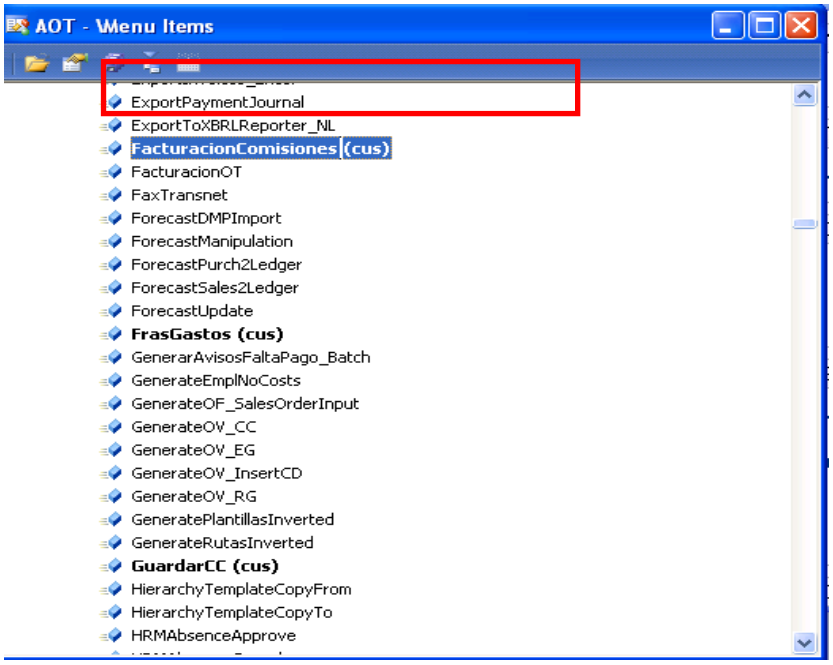
Aquest mètode és utilitzat per executar una classe. No és un mètode de la classe, sinó que, quan és vol executar una classe a través d'una crida de *menuItem* es crearà un mètode *Main* passant-li per paràmetres un objecte de tipus Args.

Aquest mètode es caracteritza perquè és un mètode estàtic. Rep un únic paràmetre de tipus Args i mai retorna cap tipus de paràmetre, és a dir, aquest mètode és del tipus Void.

Exemple:

```
void static main (args _args);
```

Una classe pot ser executada mitjançant un menuItem del tipoAction o sobre la mateixa classe, polsant "Abrir" amb el botó dret del ratolí o mitjançant la barra de eines d'Axapta que es mostra a continuació:



Il·lustració 4 Exemples de Menús Ítems

3.2.4.3. Herència

X++ implementa el que es coneix com un model d'herència simple. Una nova classe sols pot ser subclasse d'una altra classe. Si volem estendre una classe heretarem tots els mètodes i variables de la classe mare. La classe mare s'anomena superclasse. La sintaxis que determina que una classe és filla de una superclasse és la següent:

*Class **Nom de la classe filla** extends **nom de la superclasse***

A continuació, presentem un exemple que crea una nova classe que és una variant de la classe Point. Aquesta nova classe l'anomenarem 3DPoint:

```
class Point
{
        real x; // instància de la variable.
        Real y; // instància de la variable.
        ;
```

```
        New(real _x, real _y)
        { // el constructor inicialitza les variables x i y.
            x = _x;
            y = _y;
        }
    }
    class 3DPoint extends Point
    {
        real z; // la coordenada z del punt.
        New(real _x, real _y, real _z)
        {
            super(_x, _y); // inicialitza les instàncies
            z = _z;
        }
    }
```

Com podem observar, la classe 3DPoint ha afegit una nova instància de variable per a la cordenada z del punt. Les instàncies x i y són heretades de la classe Point i, per tant; no és necessari declarar-les en la classe 3DPoint.

3.2.5. Sentències bàsiques de la programació

En aquest apartat ens centrarem en conèixer a grans trets quines són les sentències bàsiques per al llenguatge X++. A part de la seva funcionalitat, també veurem com s'implementen en X++.

- Instrucció *if*. És una funció condicional bàsica que es defineix de la següent manera:

if *(expressió) instruccions [*else *instruccions]*

L'expressió es pot avaluar com a vertader o fals.

- Instrucció *Switch*: aquesta sentència condicional de bifurcació múltiple ens permet seguir diversos camins, incloent-hi un per defecte, segons el valor de l'expressió:

```
Switch (Expressió)
{
    case Expressió {, Expressió}:
        Instruccions;
        [break;]
    case...

    [default
        Instruccions;
    ]
}
```

Una operació *switch* pot substituir-se com un conjunt d'operacions *if* consecutives. L'ús de la instrucció break depèn de si es vol seguir processant els diferents camins de bifurcació quan ja se n'ha trobat un.

- Instrucció *While*. Ens permet repetir una iteració mentre es compleix una o varies condicions.

```
while ( Expressió)
    Instruccions
```

- Instrucció *Do While*: aquesta instrucció bàsicament té la mateixa funcionalitat que la instrucció *while*. L'única diferència és la seva estructura.

```
do {
{Instruccions}
```

} while (Expressió)

- Instrucció *for*. És una ampliació de la instrucció *while*. Ens permet inicialitzar la variable de control de la instrucció en la mateixa iteració.

for (Inicialització; Expressió; Actualització)
Instruccions

3.2.6. Consulta a les dades del sistema

A continuació, descriurem una sèrie d'instruccions per poder accedir a les dades que el sistema té emmagatzemades en la base de dades. Per a l'accés a aquestes dades, Axapta utilitza comandes del SQL per facilitar la consulta de les mateixes.

- *Select*: aquesta instrucció ens serveix per buscar i manipular dades de la base de dades. S'utilitzarà una variable del tipus taula la qual s'haurà de declarar abans. Aquesta instrucció ens retornarà una selecció de registres de la taula que volem consultar i la que s'indica en la instrucció.

  Les opcions de cerca que tenim per a aquesta instrucció són les que es mostren a continuació:

  - *Reverse*: els registres seleccionats es retornen de forma inversa.

  - *Firstfast*: ens serveix per accelerar la consulta del primer registre de la taula.

  - *Firstonly*: únicament ens retorna el primer registre de la taula.

  - *Forupdate*: els registres als que accedim amb la consulta podran ser modificats en utilitzar aquesta instrucció.

Les opcions de càlcul per a la sentència *select* són:

- *Sum*: per obtenir la suma dels registres seleccionats.
- *Avg*: per obtenir la mitja dels registres seleccionats.
- *Minof*: per obtenir el valor mínim dels registres seleccionats.
- *Maxof*: per obtenir el valor màxim dels registres seleccionats
- *Count*: per obtenir el número de registres de la taula seleccionada.

Per altra banda, la instrucció *select* ens permet obtenir les files seleccionades de forma agrupada mitjançant la sentència *group by*.

A més a més, també ens permet obtenir els registres de forma ordenada mitjançant la sentència *order by.*

Cal tenir en compte, que podem seleccionar registres de diferents taules que estiguin relacionades mitjançant l'ús de la instrucció *join*.

Podem definir diferents instruccions per obtenir registres de diferents taules segons la relació entre elles:

- *Join*: selecciona els registres de la taula principal que tenen registres associats a la taula relacionada.
- *Exists join*: selecciona els registres de la taula principal si existeixen registres associats a la taula relacionada.
- *Notexist join*: selecciona els registres de la taula principal que no tenen registres associats a la taula relacionada.

  - *Outer join*: selecciona els registres de la taula principal tinguin o no registres associats a la taula relacionada.

- *While select*: aquesta sentència ens permet desplaçar-nos entres els registres sense necessitat d'utilitzar la instrucció *next*.

- *Delet_From*: ens permet esborrar tots els registres que compleixen una condició.

- *Insert_Recordset*: s'utilitza per copiar des d'una o més taules en una taula específica.

- *Update_Recordset:* ens permet la manipulació de camps en un conjunt de registres en una única operació.

4. Introducció a la funcionalitat estàndard del mòdul Gestió d'Inventari

El mòdul de gestió de magatzems ens ajuda a administrar les activitats relacionades del mòdul incloent-hi la recepció de materials, l'emmagatzemament, la transferència de productes entre diferents ubicacions, ja sigui, en el mateix magatzem o en d'altres i l'enviament de productes en relació a la producció, a la compra i a les comandes de venta.

A més a més, la gestió de magatzems de AX ens permet optimitzar i simplificar el flux de treballs d'un magatzem. El sistema es pot configurar segons la forma en què una empresa vol controlar el seu inventari.

Una vegada posat en marxa aquest mòdul el sistema fa que la forma de treballar al magatzem sigui més eficient i més fàcil de manejar. També, ens proporciona la flexibilitat i el control per optimitzar els processos del magatzem d'acord a les necessitats individuals, depenent sempre, de l'empresa. Els principals beneficis del sistema de gestió de magatzems són:

- Optimització dels dissenys dels magatzems per incrementar l'eficiència.

- Obtenir una visió completa de l'inventari per ajudar a millorar el servei al client.

- Racionalitzar els processos per a la reducció dels costos.

Podem destacar d'aquest mòdul la flexibilitat que ens proporciona per definir la disposició d'un magatzem en funció de les seves necessitats canviants i, d'aquesta manera, arribar aconseguir la màxima eficàcia possible. Ens permet doncs, establir àrees d'emmagatzemament d'alta o baixa prioritat per a la col·locació òptima dels bens d'una empresa i, a més a més, definir llocs per restringir el moviment de les mercaderies. També cal tenir present, que es pot dividir un magatzem en zones per donar

cabuda a les diferents necessitats d'emmagatzemament, com per exemple: requisits de temperatura o la velocitat de rotació dels articles.

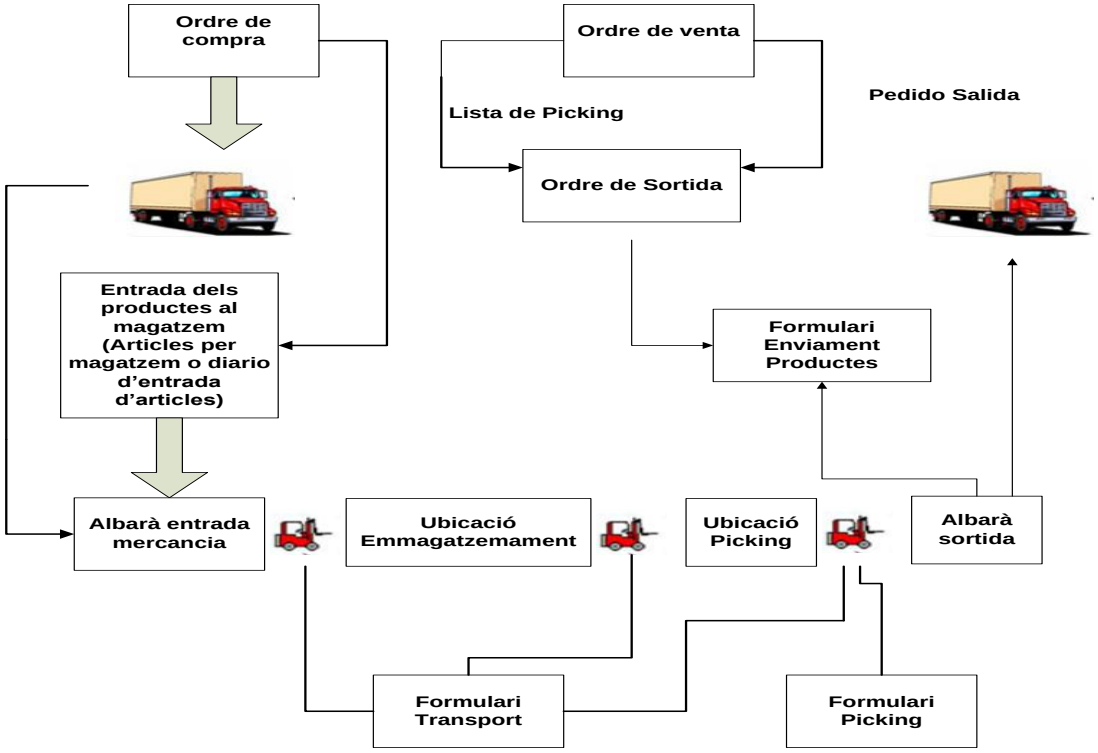
Axapta, ens ajuda a utilitzar l'espai d'un magatzem de forma eficaç, identificant automàticament, les ubicacions òptimes d'emmagatzemament per als articles en funció de les característiques dels elements d'emmagatzematge i de les zones del magatzem.

A continuació, veurem quins són els mestres principals del mòdul i els processos generals que aporta l'aplicació estàndard.

4.1. Esquema general d'Axapta

4.1.1. Esquema del mòdul d'inventari amb la resta de mòduls

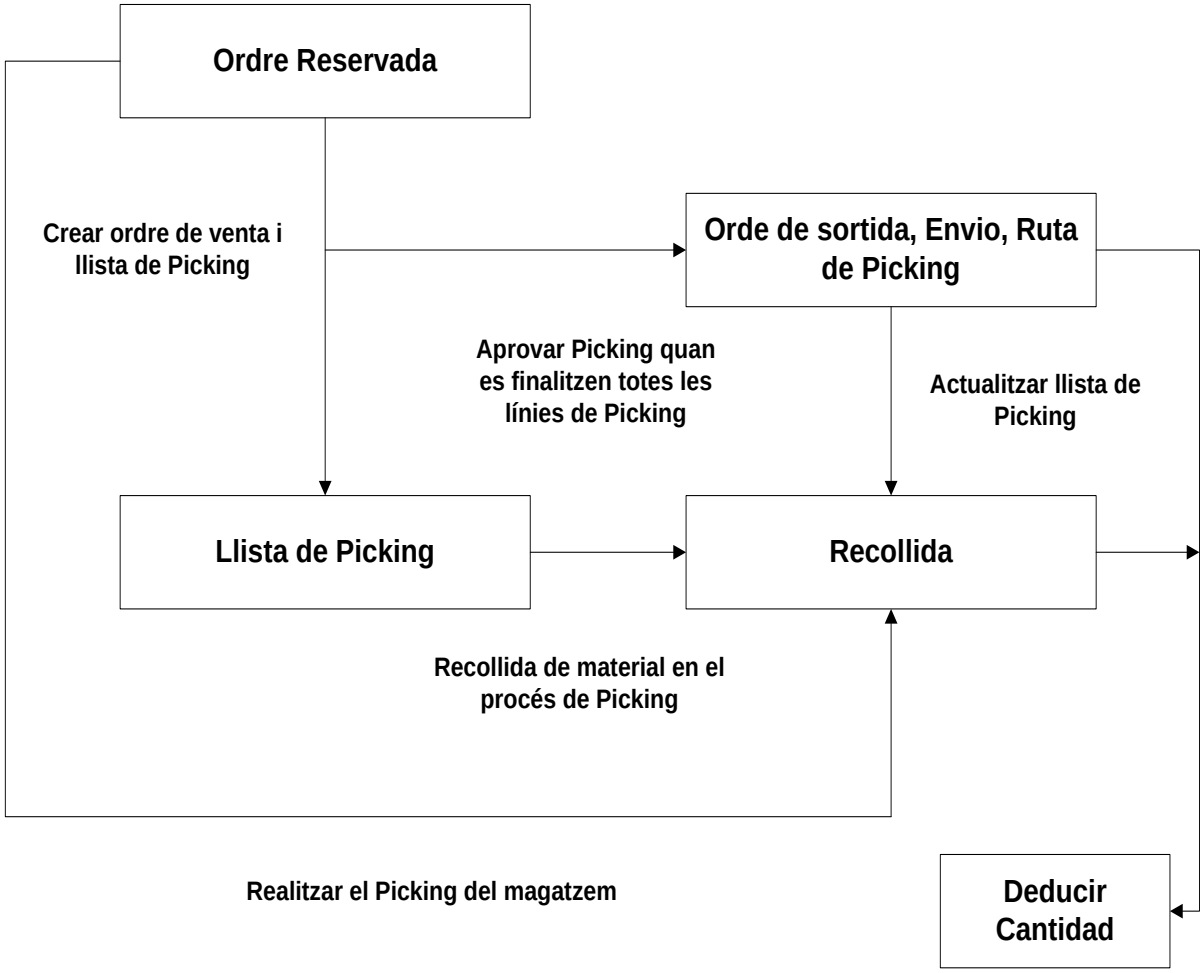
A continuació veurem quin és el flux de treball d'un magatzem en relació amb altres mòduls d'Axapta:



Il·lustració 5 Esquema mòdul inventari amb la resta de mòduls

4.1.2. Esquema del procés de Picking

En el següent esquema podem observar quin és el flux de treball per al procés de Picking i per als seus elements relacionats d'Axapta:



Il·lustració 6 Esquema mòdul inventari amb la resta de mòduls

4.2. Formularis comuns

4.2.1. Mestre d'Articles

El formulari d'articles ens permet crear i administrar les dades bàsiques dels bens d'una empresa, així com, les llistes de materials i els articles de serveis. És a dir, AX ens permet definir tres tipologies d'articles:

- Article: aquest tipus pot incloure articles acabats, semiacabats o matèries primes. No és possible vincular altres articles del tipus llista de material a aquest tipus d'articles.

- Llista de material: els articles tipus llista de material són aquells que es composen d'altres articles també del tipus material o bé d'articles finals o de servei.

- Article de Servei: aquests tipus d'articles són la resta d'articles que no es poden incloure en cap dels tipus anteriors. El tipus d'article de servei, com el seu propi nom indica, s'utilitzen per donar d'alta en el mestre d'articles tots aquells serveis que una empesa pot portar a terme i pot rebre. Un exemple d'article de servei podria ser la llum, l'aigua, el telèfon, etc.

La creació d'un article comporta haver d'informar una sèrie de camps bàsics del formulari que s'expliquen a continuació:

- Camps que obligatòriament s'han d'informar:

    - Número de l'article: és un codi únic definit per l'usuari el qual s'assigna a la creació d'articles, és a dir, podríem definir el número de l'article com el camp al que fem referència per a la seva gestió. Es recomana no utilitzar caràcters especials així com espais en blanc. Aquest camp es pot configurar, a fi que, el sistema al crear un nou article assigni de forma automàtica un codi d'article.

- Grup d'articles: aquest camp ens serveix per administrar l'inventari, dividint els articles en grups en funció de les seves característiques. També es pot configurar número de comptes contables per a les transaccions contables que es generen de forma automàtica i, generades en relació a les transaccions de recepció i emissió del grup d'articles en l'inventari.

- Grup de model d'inventari: un article ha de contenir un grup de model d'inventari, ja que, aquest grup ens permet determinar com es controlen i es manipulen els articles quan es reben o s'emeten. També determinen com es calcula el seu consum. Un grup de models es pot associar a diferents articles, fet que, facilita el manteniment perquè molts articles es controlen amb la mateixa configuració. Axapta ens permet valorar el consum de l'inventari per diferents mètodes:

  - FIFO: la valoració de les sortides es fa en funció de les primeres entrades.

  - LIFO: la valoració de les sortides es fa en funció de les últimes entrades dins d'un interval de temps.
  - Data LIFO: la valoració de les sortides es fa en funció de les últimes entrades.

  - Mitja ponderada: la valoració de les sortides es fa en funció del cost mig de les entrades dins d'un interval de temps.

  - Mitja ponderada a data: la valoració de les sortides es fa en funció del cost mig de les entrades.

  La diferència entre els mètodes de valoració a data i els que no es fan a data és la següent: els mètodes de valoració que no són a data tenen en compte les entrades i les

sortides dels productes en un interval de temps que coincideix amb els tancaments de període.

- Grup de dimensions d'inventari: les dimensions d'inventari ens indicaran el detall d'emmagatzemament dels articles i la forma en què aquests es manipularan. Els articles que manipularem de la mateixa manera s'assignaran al mateix grup de dimensions d'inventari. La relació amb cada article s'introdueix en el mestre d'articles.

  El formulari de creació de grups de dimensions d'article es divideix en tres parts:

  - En la primera part, introduirem el codi i la descripció del grup de dimensions de l'inventari.

  - En la segona part, activarem les dimensions de l'article.

  - En la tercera part, activarem les dimensions d'emmagatzemament.

  Les dimensions d'article s'utilitzen per identificar un article que té diversos atributs, per exemple: model, color i grandària. En lloc de crear diferents articles per a cada combinació d'atributs podem crear un únic article i les seves possibles combinacions d'atributs.

  Les dimensions d'emmagatzemament disponibles en AX són les següents:

  - Magatzem: és la ubicació de depòsit o d'emmagatzemament, per exemple: un local o edifici.

- Ubicació: és la ubicació de depòsit o d'emmagatzemament específica, per exemple: un passadís, una prestatgeria, una balda o un lloc concret anomenat "Hueco2" en AX que és el lloc on finalment dipositarem l'article.

- Número de Lot: és un identificador d'un grup d'articles relacionats.

- Número de Sèrie: és un identificador únic de l'article.

- Pallet ID: és un identificador d'un grup d'articles dins d'un mateix pallet.

Per a cada grup de dimensions d'article, podem activar entre les disponibles, les dimensions que volem en funció del grau de detall d'emmagatzemament que es consideri necessari per gestionar el producte.

- Altres camps que fan referència als articles però, no són obligatoris per a la seva creació:
  - Nom: ens permet informar del nom de l'article. Aquest apareixerà per exemple: al document de la factura o de l'albarà.

  - Pes: ens permet indicar quin és el pes del producte.

  - Volum: ens permet indicar el volum del producte.

  - Grup d'impostos d'article: ens permet especificar quin serà el percentatge d'Iva que s'aplicarà a l'article en el cas de realitzar una compra o una venta.

- Preu de compra: permet especificar un preu de compra per l'article.

- Preu de cost: ens permet especificar o veure quin és el preu de cost de l'article. Aquest camp s'actualitza al realitzar el tancament d'inventari, procés que s'explicarà més endavant.

- Preu de venta: permet especificar el preu de venta del producte.

- Unitats i conversió d'unitats: des del formulari d'unitats podem introduir les diferents unitats de mesura dels articles, com per exemple: unitat, peces, caixes, kilograms, centímetres, etc. Així com, la conversió genèrica o específica d'unes unitats a altres. Exemple de conversió:

  Escollida una unitat de (centímetres), introduirem una quantitat (per exemple 100) que pel factor de conversió (*), ens donarà la unitat a la que volem convertir (metres). El resultat seria: 100 cm = 1 m.

  AX ens permet tenir diferents unitats per a un mateix article segons:

  - Unitat de l'inventari: ens permet especificar quina és la unitat de l'inventari pel recompte dels productes, per exemple: una unitat d'inventari podria ser Kilograms, unitats, etc.

  - Unitat de compra: ens permet especificar quina és la unitat de compra dels productes, per exemple: Kilograms, unitats, etc. La unitat de compra no té perquè ser igual a la de l'inventari.

- Unitat de la venta: ens permet especificar quina és la unitat de venta dels productes, per exemple: Kilograms, unitats, etc. La unitat de venta no té perquè ser igual a la de l'inventari o a la de la compra.

- Classificació ABC: la classificació ABC és utilitzada pel control d'inventaris i, consisteix en classificar els productes amb el tipus A, B o C segons el criteri i el percentatge establert. Els materials es poden classificar pel valor d'inventari, pel valor de venta, pel valor de consum, per la quantitat consumida o pel criteri desitjat. Es tracta doncs, que segons el criteri que s'utilitzi els materials del tipus A siguin els més importants, els del tipus B els entremitjos i els del tipus C els menys importants. Per exemple: el 10% dels productes representen el 60% de les compres de l'empresa i, per tant; aquesta és la zona A, el 40% dels productes representen el 30% i aquests productes són els que estaran ubicats en la zona B, la resta el 50% dels productes que representen el 10% de les compres són els productes del tipus C. Aquests valors són arbitraris, ja que, cada empresa té les seves particularitats i si algú decideix utilitzar aquests criteris ha de ser conscient de la realitat de l'empresa. No solsament s'ha de pensar en els costos sinó que és també important veure altres criteris, la qual cosa , és sens dubte la principal dificultat d'aquest tipus d'anàlisi.

  Cada empresa utilitza diferents productes i cadascun d'ells tenen les seves pròpies característiques. Per tant, cada producte necessita una gestió particular depenent de la seva importància en els processos de la companyia i de les possibilitats d'adquisició. Pensar que tots els productes s'han de controlar de la mateixa manera constitueix una visió limitada de la realitat que implica un desgast i sobrecostos innecessaris.

AX permet 4 tipologies de classificació ABC:

- Per valor: en funció del valor de l'inventari, és a dir, fan referència al valor que representen en l'inventari. Per exemple:

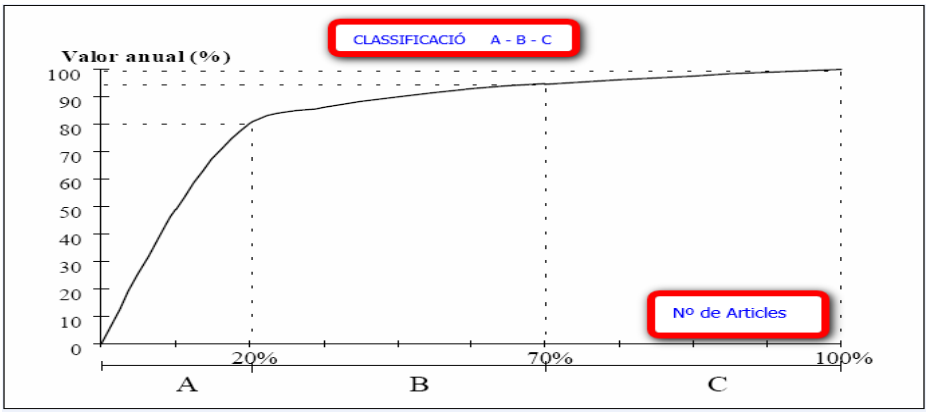
  - Els articles A constitueixen el 20 % de l'inventari total, per tant, representen el 80% del valor total de l'inventari.

  - Els articles B constitueixen el 50% de l'inventari total i representen el 15% del valor total de l'inventari.

  - Els articles C constitueixen el 30% de l'inventari total i representen el 5% del valor total de l'inventari.

Un article es classifica dins del grup A perquè té un elevat cost, molts moviments anuals o bé per ambdues raons. Evidentment, segons sigui per una raó o per l'altra el seu control serà d'un tipus o d'un altre.

De forma gràfica:



Il·lustració 7 Exemple de classificació ABC

- Per marge: ens permet classificar el nombre d'articles a partir del valor d'inventari actual que es calcula en funció de les transaccions d'inventari.

- Per ingressos: ens permet classificar el nombre d'articles a partir de la venta de totes les emissions d'articles en un interval de dates específiques. El càlcul es basa en les transaccions de l'inventari.

- Per rotació: ens permet classificar el nombre d'articles a partir del cost de rotació de l'inventari calculat. Aquest cost es basa en les transaccions d'inventari. Els resultats s'insereixen a la taula d'articles en el camp codi ABC corresponent al model seleccionat, on A = al percentatge més alt, B = al percentatge mig i C = al percentatge més baix.

- Consultes del mestre d'articles:

  - Disponible: ens permet veure la disponibilitat de l'article seleccionat en funció de les dimensions d'inventari, o sigui, podem veure el total disponible per número de lot, per magatzem, per ubicació, per color, etc.

  - Transaccions: ens permet consultar quines han estat les transaccions realitzades pel producte seleccionat, és a dir, podem veure per exemple: quines compres / ventes s'han realitzat, així com, la quantitat de les mateixes, l'import, etc.

  - Ubicació del producte al magatzem: podem consultar i a la vegada configurar quina serà la ubicació d'emmagatzemament pel producte, així com; indicar el magatzem, el passadís, prestatgeria, etc.

4.2.2. Mestre de llistes de materials

Una llista de material o BOM (*Bill Of Material*) és el conjunt de components que constitueixen un article del tipus llista de material (L.MAT), aquests components poden ser articles del tipus article o bé articles del tipus llista de material.

Les llistes de material poden definir-se independentment dels articles a través del mestre de llistes de material; per tant, la llista de material podria estar relacionada amb diferents articles tipus L.MAT.

Utilitzarem aquest formulari per a crear, configurar i modificar llistes de materials. Tota llista de material estarà formada per una sèrie de matèries primes o d'altres llistes de materials. Per exemple: una llista de material podria ser un cotxe, on el motor seria una llista de material del producte final i les llantes una part de la matèria prima del cotxe.

Un llista de material pot contenir 1 o N versions de llistes de materials. En l'exemple del cotxe la llista de material seria el cotxe i les versions podrien ser en funció del color:

- Versió 1: cotxe blau
- Versió 2: cotxe negre
- ...
- Versió N: cotxe blanc.

A més a més, tota versió de llista de material pot tenir unes dates de validesa temporal, és a dir, poden tenir vigència.

Les llistes de materials poden ser configurables. La mateixa llista de material tindrà diferents configuracions, en funció, que escollim uns components o uns altres per a la creació de la llista de material.

Creem un nou registre i introduïm el nom de la llista de material. Des d'aquest formulari anirem introduint els components de la llista de material, que com ja s'ha comentat anteriorment, poden ser article o altres llistes de materials. També podríem utilitzar el dissenyador per compondre la llista de material. Introduirem el codi de l'article, la seva configuració o bé el grup de configuració i la quantitat en unitats físiques, del component de la llista per a cada unitat de llista de material en el camp quantitat.

Una vegada hem introduït aquests camps procedirem a aprovar la llista de material. Una vegada està aprovada podrem activar únicament una versió per a la llista de material. Cal recordar doncs, que ens les versions de llistes de materials, en el cas, de tenir més d'una versió podem especificar un període de validesa per aquesta.

4.2.3. Rutes de Picking

Utilitzarem aquest formulari per gestionar les rutes de Picking. S'utilitza per al Picking consolidat, on la ruta de recollida de material s'ordena per ubicació, per exemple: la més propera és la més prioritària i la resta es prioritza segons la proximitat.

Des d'aquesta pantalla es seleccionarà la ruta de Picking que es vol iniciar en funció de les comandes de ventes que estiguin preparades per iniciar el Picking. Un cop seleccionada la ruta de Picking s'haurà d'iniciar i, a la vegada, crearem un Pallet de Picking pel transport de les mercaderies.

En la part inferior d'aquest formulari seleccionarem les línies que estiguin preparades per realitzar el Picking i les aprovarem per indicar que ja es poden preparar.

Finalment, per aquelles línees de la comanda de venta que s'hagi fet el Picking es podrà realitzar també la reposició per a la ubicació de Picking del producte que s'ha servit, en cas de ser necessari.

4.2.4. Magatzems

Utilitzarem aquest formulari per a la creació de magatzems per al nostre inventari.

Al crear un nou magatzem AX ens obliga a introduir el tipus de magatzem creat, aquests poden tenir les següents classificacions:

- Predeterminat: magatzem genèric que pot tenir associat un magatzem de quarantena i també un de trànsit.

- Quarantena: aquests magatzems s'utilitzen per deixar els productes en quarantena per a futures inspeccions o revisions del material entrat al magatzem predeterminat.

- Trànsit: s'utilitza quan es serveixen articles, per poder introduir-los en aquest magatzem i tenir un control de tot el que s'ha enviat o s'ha tret del nostre magatzem predeterminat. Per últim, quan el producte arriba al seu destí es treu del magatzem de trànsit.

En tot magatzem que creem podem indicar quina és la seva ubicació predeterminada per a la entrada i per a la sortida dels productes. A més a més, podem veure mitjançant una estructura en arbre les ubicacions del magatzem i com està estructurat.

4.2.5. Àrees d'emmagatzemament

Utilitzarem aquest formulari per designar seccions del magatzem per a un inventari amb característiques similars.

Crearem una nova àrea d'emmagatzemament, en la qual, indicarem el codi, el nom de l'àrea i, a més, haurem d'especificar el tipus d'àrea d'emmagatzemament. AX ens permet identificar dos tipologies d'àrees d'emmagatzematge on cadascuna d'elles tenen ubicacions connectades, aquestes són:

- Àrea d'entrada: s'utilitza amb relació amb la ubicació d'emmagatzemament dels articles quan es reben en l'inventari. Les àrees d'emmagatzemament es divideixen en zones d'emmagatzematge. Un exemple de sistema d'àrea d'entrada és: la classificació ABC explicada en el mestre d'articles.

- Àrea de Picking: s'utilitza per determinar les rutes de Picking. Les línies de Picking que constitueixen una ruta de Picking estan formades pels articles emmagatzemats en les ubicacions dins de la mateixa àrea de Picking.

Finalment, si es vol diferenciar el magatzem en àrees, hem de definir per a cada àrea quines són les ubicacions que comprenen cadascuna de les àrees definides.

4.2.6. Zones d'emmagatzemament

Utilitzarem aquest formulari per definir zones d'emmagatzemament en l'inventari, i així, d'aquesta manera poder ordenar i classificar per prioritat, la selecció de les àrees d'emmagatzemament per l'inventari entrant.

A més a més, aquest formulari ens permet tenir per àrees fins a N zones. En cadascuna d'elles podem especificar quins són els articles que hi han emmagatzemats, és a dir, podem definir per zones quins articles volem emmagatzemar-hi. Per exemple: podríem tenir la zona de fred on hi emmagatzemaríem el productes que s'hagin de conservar en temperatures fredes.

4.2.7. Ubicacions

Utilitzarem aquest formulari per gestionar les ubicacions d'inventari en un magatzem. Es pot identificar la ubicació d'un article mitjançant les coordenades: passadís, prestatgeria, balda i lloc d'ubicació final.

AX ens permet la possibilitat de generar les ubicacions del magatzem mitjançant un assistent. Aquest assistent ens agilita i ens dóna molta facilitat per a la creació d'ubicacions. AX diferencia els diferents usos de les ubicacions en:

- Moll d'entrada: identifica les ubicacions en les que es reben i es registren els articles.

- Moll de sortida: identifica les ubicacions en les quals es recopilen els articles abans de distribuir-los.

- Ubicació d'emmagatzemament: identifica les ubicacions en què s'emmagatzemen els articles fins que són traslladats a un moll de sortida o s'utilitzen per aprovisionar una ubicació de Picking.
- Ubicació de Picking: identifica les ubicacions en les quals els articles tenen ubicacions de Picking fixes. S'utilitza per a quantitats d'enviament inferior a pallets sencers. En aquestes ubicacions normalment es solen emmagatzemar "bultos" (unitats per separat).

En tota ubicació podem especificar el número màxim de pallets que es pot emmagatzemar, així com, les dimensions de la ubicació com per exemple: altura, fondària, etc.

A més a més, des d'aquí també podem especificar a quina àrea d'emmagatzematge pertany la ubicació creada.

4.3. Processos del mòdul

4.3.1. Diaris d'inventari

Els diaris d'inventari permeten realitzar manualment ajustos en els nivells d'inventari, recomptes, llista de materials tot hi que no hi hagi un procés de producció, transferències d'inventari, registre de pèrdues i guanys d'inventari.

A continuació explicarem el funcionament de cadascun dels diaris d'inventari.

4.3.1.1. Diari de moviments

Mitjançant el diari de moviments d'Inventari podrem registrar rebuts i emissions, o el que és el mateix, realitzar ajustos d'inventari a través d'altes o baixes de productes.

Quan introduïm la quantitat en aquest diari podem diferenciar clarament de quan es tracta d'una alta o d'una baixa, per exemple:

- La introducció de la quantitat positiva, és un rebut, o sigui, una alta d'existències.

- La introducció de la quantitat negativa, és una emissió, és a dir, una baixa en l'inventari.

És important tenir en compte que aquest tipus de diari necessita que especifiquem una conta de contrapartida[5] per a què el sistema faci l'assentament contable[6] de forma correcta.

[5] Conta de contrapartida, és la conta que farà que l'assentament contable es compensi. Tot assentament contable té que quadrar.

[6] Assentament contable, inscripció en un llibre de comptabilitat d'un esdeveniment identificat que afecta a una empresa econòmica o financerament i que es troba degudament quantificat en unitats monetàries [Assentament].

Si és per a beneficis de l'inventari la conta de contrapartida tindrà com a concepte "Beneficis d'inventari". En canvi, si és per a pèrdues d'inventari anirà a una contrapartida amb el concepte de pèrdues d'inventari.

4.3.1.2. Diari de pèrdues i guanys

El diari de pèrdues i guanys serveix igualment per realitzar ajustos d'inventari de forma manual. Podem actualitzar disminucions d'inventari (pèrdues) introduint quantitats negatives d'article i, actualitzar increments d'inventari (guanys) introduint quantitats positives d'article.

La diferència que té aquest tipus de diari si el comparem amb el de moviments, radica en què, en aquest diari no hem d'introduir la conta de contrapartida, Axapta utilitzarà les comptes d'inventari del grup d'articles.

4.3.1.3. Diari de transferències

Mitjançant el diari de transferències podem realitzar canvis en les dimensions dels articles sense que es produeixi cap tipus de comptabilització. Podríem per exemple: transferir quantitats d'un article des d'un magatzem a un altre per evitar retards en el subministrament d'aquest últim.

La quantitat introduïda en la línia del Diari de Transferència ha de ser negativa donat que es transferirà a una altra dimensió.

Finalment, podem dir que aquest diari mai produeix assentaments contables.

4.3.1.4. Diari de llistes de materials

A través del diari de llistes de materials podem realitzar processos per als casos en què no s'utilitzi el mòdul de producció:

- Donar d'alta un article del tipus llista de material i de baixa els seus components, tant manualment com automàticament

- Donar d'alta els components d'una llista de material i donar de baixa l'article de la llista de material que tenia anteriorment, tant manualment com automàticament.

4.3.1.5. Diari de recompte

El diari de recompte d'AX ens ajuda a comprar els estocs teòrics i reals, és a dir, el sistema mostrarà l'estoc teòric o les existències d'un article que té guardades en el sistema perquè les comparem amb l'estoc real contat i, introduït pel personal del magatzem. Axapta calcula automàticament les diferències entres els dos tipus d'inventari i realitza els apunts contables dels ajustos. Cal tenir present, que només introduint l'article en aquest diari a una data concreta el sistema informa automàticament de l'estoc disponible a la data especificada per a l'article que s'introdueix a la línea del diari.

4.3.2. Ajustos d'inventari i tancaments

4.3.2.1. Tancament d'inventari

El tancament d'inventari realitza ajustos i liquidacions per calcular els consums d'articles, en funció del mètode de valoració seleccionat en el grup de models d'inventari dels articles.

Els tancaments d'inventari s'han de realitzar perquè el valor de cost financer d'una emissió d'un article no és el definitiu fins que no es produeix el tancament d'inventari. Una vegada produït el tancament, la transacció d'emissió i les corresponents rebudes es tanquen i no es tornaran a calcular, amb la qual cosa, el rendiment del sistema millora.

4.3.2.2. Recàlcul d'inventari

El recàlcul genera un tancament d'inventari "provisional" amb la finalitat d'obtenir les millors estimacions sobre els costos i el consum d'articles a una data determinada.

Aquest procés compleix amb la mateixa funció de tancament d'inventari, és a dir, ajustar el consum d'articles segons el model d'inventari seleccionat. No obstant això, és diferent d'un tancament real, ja que:

- No tanca l'inventari.
- No tanca les transaccions d'inventari.
- Es pot executar en una agrupació d'articles.
- Es pot executar a qualsevol data des de l'últim tancament d'inventari.

El recàlcul es pot utilitzar diàriament o setmanalment per calcular el valor de consums d'articles més precisos. Així doncs, es podria utilitzar quan hi ha grans períodes entre els tancaments definitius.

4.3.2.3. Ajustos d'inventari

Podem realitzar dos tipus d'ajustaments:

- Ajustos de disponible: Podem ajustar el valor del cost de l'inventari disponible per a cada article en funció de les dimensions seleccionades. Prèviament, és necessari realitzar el tancament d'inventari per als articles que es vulgui realitzar l'ajust de disponible.
- Ajustos de transaccions: Podem ajustar el valor de cost d'una transacció en concret, per exemple: per a una recepció d'inventari amb estat comprat. Per realitzar aquest tipus d'ajustaments no és necessari haver realitzar el tancament prèviament.

4.3.3. Transferir comandes entre magatzems

El procés de transferir comandes entre magatzems ens serveix per a aquelles empreses que tenen més d'un magatzem. En aquestes empreses cada magatzem realitza la seva gestió de compra i de venta per separat però sempre de les mateixes referències de productes, és dir, compren i venen els mateixos productes. Aquest procediment ens permet transferir comandes en l'hipotètic cas que en un magatzem no es pugui servir a un client a la data ABC que marca la venta. Aleshores, el sistema mitjançant aquesta funcionalitat ens permet transferir la comanda del magatzem que no pot servir a l'altre per a què aquest en sigui el responsable de servir al client els productes demanats.

4.3.4. Codi d'ordenació

El procés de codi d'ordenació permet ordenar de forma automàtica les ubicacions del magatzem escollit. Aquesta funcionalitat ordena les ubicacions de menor a major pes, començant pel passadís 1 del magatzem i tenint en compte les prestatgeries del mateix fins al passadís N del magatzem. De manera que, tindrem ordenat per passadís totes les ubicacions del nostre magatzem. Aquesta ordenació és utilitzada en el procés de Picking, de forma que, el procés agafarà els productes en l'ordre del camp d'ordenació de les ubicacions, sempre de menor a major número. Exemple d'ordenació d'ubicacions en funció de la nomenglatura de les mateixes:

- Axapta ens permet definir el format de les ubicacions en funció del passadís, prestatgeria, balda i lloc, seguint sempre aquest ordre. La ubicació sempre començarà identificada pel passadís on es troba, seguida de la prestatgeria, balda i lloc. Podem tenir el següent format per a les ubicacions:

    - 01-1-01-1 : significa passadís 01, prestatgeria 1, balda 01 i lloc 1 de la ubicació.

- 01-2-01-1: significa passadís 01, prestatgeria 2, balda 01 i lloc 1 de la ubicació.

- 02-1-01-1: significa passadís 02, prestatgeria 1, balda 01 i lloc 1 de la ubicació.

- 02-2-01-1: significa passadís 02, prestatgeria 2, balda 01 i lloc 1 de la ubicació.

Si llencéssim el procés per a aquestes ubicacions Axapta les ordenaria de la següent manera:

Ubicació	Camp Ordenació
01-1-01-1	1
01-2-01-1	2
02-1-01-1	3
02-2-01-1	4

4.3.5. Classificació ABC

Axapta ens permet a través d'aquest procés classificar els productes de la seva empresa, de forma automàtica i en funció del model de classificació que esculli.

Primerament, hem d'introduir l' interval de dates per al càlcul. En el camp interès podem introduir el percentatge d'interès que afecta al cost de manteniment de les existències. En els camps A, B, C hem d'introduir els percentatges de classificació per a cada grup tenint en compte que tots ells han de sumar el 100%.

En el camp model ABC hem de seleccionar un dels quatres models de selecció que ofereix AX i que tornem a recordar:

- Valor
- Ingressos
- Marge
- Rotació

Els resultats del procés s'introdueixen a la taula d'articles, en els camps corresponents, en funció del model de classificació seleccionat.

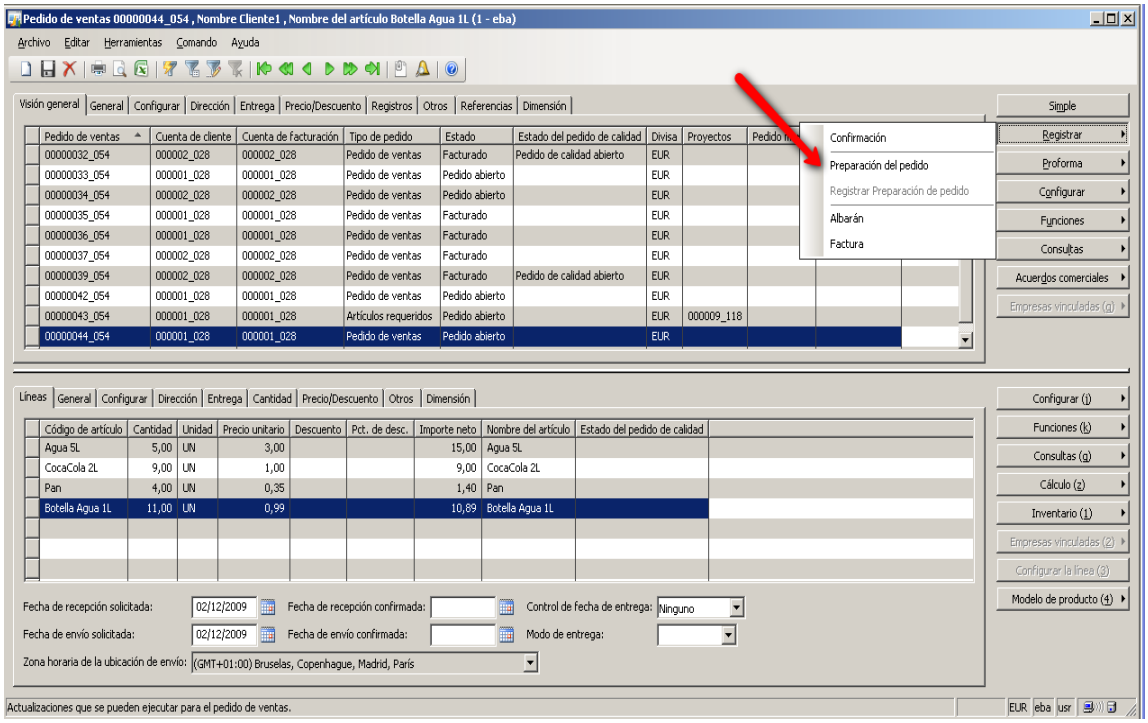
4.3.6. Procés de Picking i reposició

A continuació, es pretén explicar la funcionalitat estàndard que ofereix el sistema per a la realització del Picking i la reposició d'un magatzem. D'aquesta manera, podrem entendre el perquè s'ha decidit realitzar un altre algorisme de Picking i de reposició per al sistema AX, quan aquest ja en conté un.

En primer lloc i, com a fet essencial cal realitzar una comanda de venta per servir una sèrie de productes al client. Després de realitzar la comanda és quan ens centrarem en com servir al client en el menor temps possible per a la preparació de la comanda. Cal doncs, un procés òptim per a la realització del Picking i per a la preparació de les comandes en el menor temps possible.

Una comanda de venta en Axapta es composa d'una capçalera on s'informa de les dades del client; així com, del codi del client que tenim en el sistema, del nom del client, de les formes de pagament del client, dels mètodes d'entrega de la mercaderia, etc. I per altra banda, està formada per una sèrie de línies on tenim en cadascuna d'elles la informació dels productes a servir, el codi del producte que tenim al sistema, el nom del producte, la quantitat que cal servir, el preu de l'article, l'import, la data d'entrega i de l'enviament de la mercaderia, etc.

Un vegada tenim feta la comanda de venta per al client procedim a realitzar la preparació de la comanda. Aquest fet, implica l'inici del procés de Picking perquè un cop registrada la preparació de la comanda el sistema detecta que aquesta comanda està en estat de selecció i, llesta per a la realització del Picking. L'acció "registrar la preparació del comanda" la trobem en els comandes de venta:



Il·lustració 8 Formulari comanda de venta

El fet de registrar la preparació del comanda, Microsoft Dynamics AX, genera una llista de selecció que ens ajuda a saber quins són els articles que cal servir al client. Aquests articles apareixen en el llistat anomenat llista de selecció de forma ordenada, segons la ruta més eficient. Pel sistema la ruta més eficient és aquella que té la ubicació amb el numerador més baix i així successivament. El sistema tria les ubicacions en ordre ascendent, és a dir, de menor a major independentment de les característiques del producte com podria ser: el pes, el volum, etc.

A més a més, al registrar aquesta preparació el sistema dóna per defecte en la comanda un número de ruta de Picking per a la seva realització. Aquest també el podem observar en la llista de selecció que genera Microsoft Dynamics AX o també anomenat informe de preparació de comandes. Aquesta ruta de Picking és la que ens dona l'ordre de recollida dels productes tal i com s'ha explicat anteriorment.

A continuació, podem veure l'exemple d'aquest llista que treu Microsoft Dynamics AX:

Cliente3

Preparación del pedido

Referencia .....: Pedido de ventas
Número .....: 00000066_054
Fecha de activación .....: 26/04/2010 17:58:54
Fecha/hora final .....:
Página .....: 1 de 1
Modo de entrega .....:
Condiciones de entrega .....:
Estado de manipulación .....: Activado

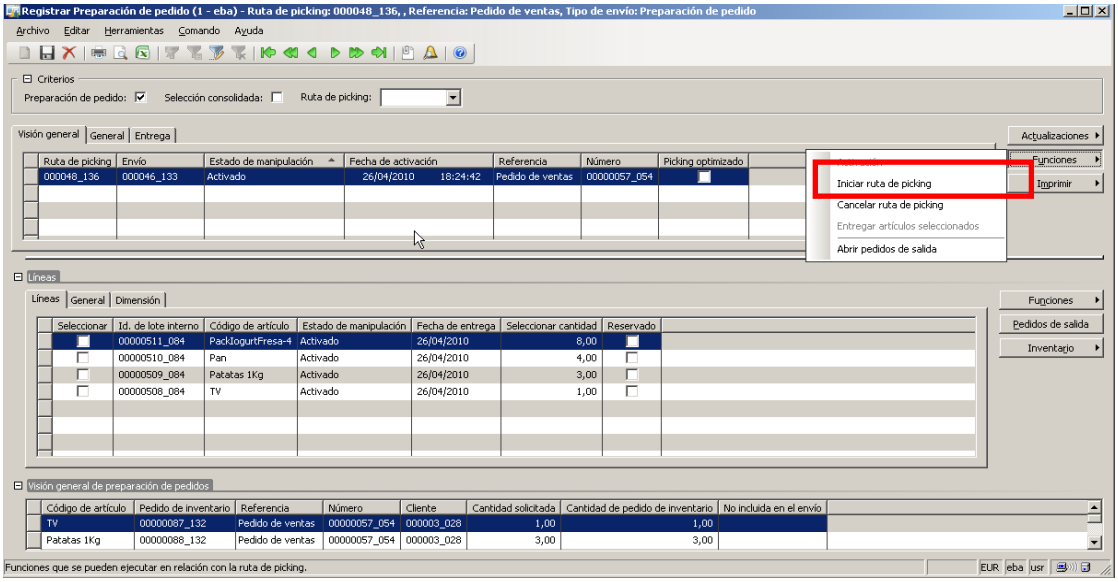
000047_136

Ruta de picking .....: 000047_136

Código de artículo	Estado de manipulación	Descripción	Cantidad de pedido	Unidad de pedido	Fecha de entrega	Cantidad	Seleccionado
PackIogurtFresa-4	Activado	Pack Iogurt Fresa	8,00	UN	26/04/2010	8,00	
Pan	Activado	Pan	4,00	UN	26/04/2010	4,00	
Patatas 1Kg	Activado	Patatas 1Kg	3,00	UN	26/04/2010	3,00	
TV	Activado	TV	1,00	UN	26/04/2010	1,00	

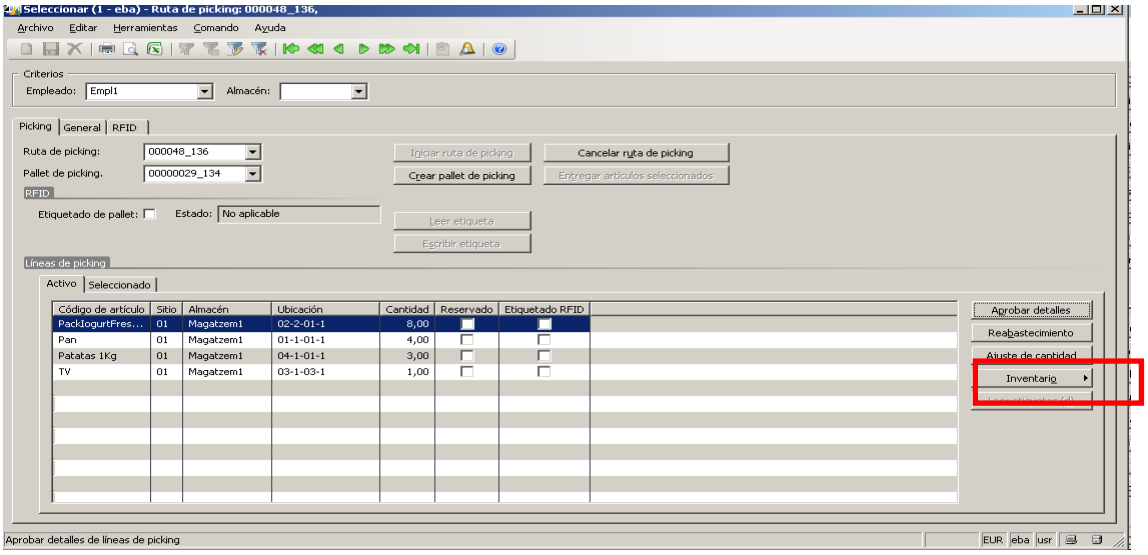
Il·lustració 9 Informe llista de selecció de productes

Quan ja tenim la preparació de la comanda feta hem d'indicar a l'usuari del magatzem (l'encarregat de servir la comanda) que ja pot començar amb la recollida dels productes. Aquest procés en Axapta s'anomena "Iniciar Ruta de Picking" i es realitza des del formulari Registrar preparació de la comanda.



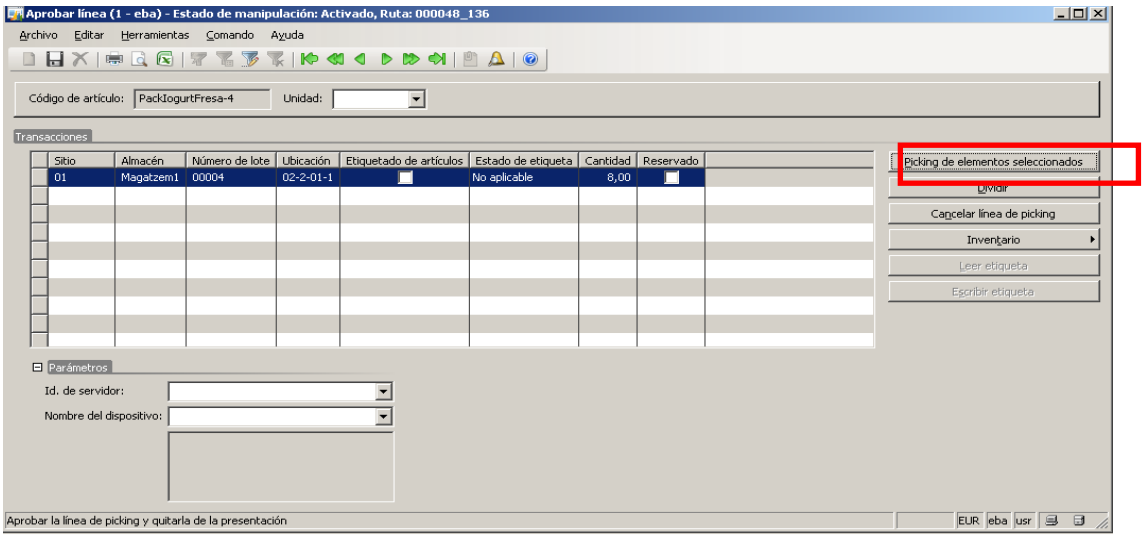
Il·lustració 10 Formulari registrar preparació comanda

Un cop iniciada la ruta de Picking cal comprovar que hi hagi suficientment estoc disponible per a la recollida de material en les ubicacions corresponents per al Picking i, realitzar el Picking de cadascuna de les línies de la comanda de venta. Per a la realització del Picking hem d'anar al formulari Rutes de Picking i aprovar cadascuna de les línies de la comanda de venta. D'aquesta manera, es considera que ja s'ha realitzat el Picking dels materials i que el següent pas seria passar a la preparació dels materials per a servir-los al client.



Il·lustració 11 Formulari Ruta de Picking

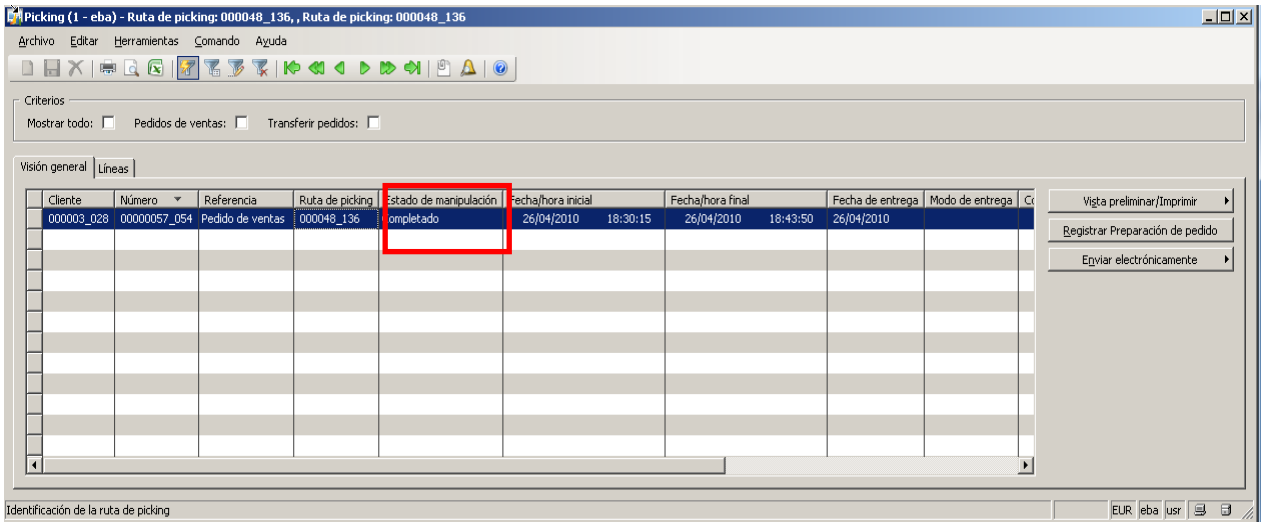
Si aprovem el sistema obre un altre formulari on procedirem a realitzar el Picking dels elements seleccionats:



Il·lustració 12 Formulari aprovació línies de Picking

En el moment que tenim tots els elements "seleccionats" el procés de Picking ha finalitzat. Tots els productes es troben al motlle de sortida, llestos per ser preparats i enviats al client.

Podem observar doncs, que el Picking ha finalitzat mitjançant la consulta des del formulari de Picking.



Il·lustració 13 Formulari Picking

Podem veure doncs, com el Picking ja s'ha acabat. A més a més, aquest formulari ens proporciona la data i l'hora d'inici en què es va iniciar el Picking i, la data i l'hora de finalització del procés. És important remarcar que és l'usuari del magatzem qui inicia el Picking i qui el finalitza. D'aquesta manera podem saber en temps real quan s'ha trigat en realitzar el Picking dels elements de la comanda.

A més a més, de la selecció dels productes cal tenir en compte el procés de reposició de la mercaderia en les ubicacions de Picking. Aquest és un procés que es realitza manualment des del formulari Rutes de Picking. Cal tenir en compte, que Axapta solament respon les ubicacions de Picking mitjançant pallets de Picking, és a dir, no podem aprovisionar una ubicació de Picking amb un únic element o variïs productes.

Finalment, podem concloure que el procés de Picking i de reposició de Axapta és molt escarit, ja que, únicament ordena la ruta de Picking (recollida dels elements) en funció del nom de la ubicació (com s'ha comentat anteriorment) i fent memòria, ordena de forma ascendent sense tenir en compte les característiques del producte. Per tant, es decideix realitzar un nou algorisme que calculi el camí òptim de recollida de material tenint en compte el pes i el volum dels materials, de forma que, quan recollim els productes els puguem apilar els uns damunts dels altres sense cap inconvenient i sense por de què es facin malbé o es trenquin.

5. Requisits de l'algorisme a implementar

5.1. Anàlisis de l'algorisme

En aquest projecte, l'algorisme que s'implementa està pensat per millorar el procés de Picking i de reposició que el mateix sistema ERP incorpora en l'aplicació.

Els objectius principals d'aquest procés consisteixen, bàsicament, en l'obtenció d'una ruta òptima de recollida del material per a les comandes que s'han de servir als clients, en el mínim de temps possible. La ruta de recollida del material s'obtindrà en funció d'una sèrie de criteris que descriurem en aquest capítol. A més a més, també es pretén aconseguir una millora en la col·locació dels productes i en el mitjà de transport utilitzat en la seva recollida, a fi d'aconseguir, que aquests no és deteriorin en el procés de preparació de la comanda.

Cal destacar, que una de les característiques més importants que s'ha tingut en compte per realitzar aquest algorisme, és la confecció d'un disseny recursiu per al càlcul dels possibles camins de recollida dels productes al magatzem. Això és de vital importància, ja que, és necessari optimitzar el temps de resposta de l'algorisme per fer-lo el més eficient possible.

La recursivitat és un procediment o funció que fa referència a si mateix dins de la seva definició. A la pràctica, la recursivitat es dóna quan un procediment o funció produeix crides successives fins arribar a un estat de no generació de més crides successives dins l'algorisme. Per exemple: en el cas de l'algorisme a implementar, si tenim N productes per preparar, vol dir que necessitarem passar per N ubicacions dins del nostre magatzem per a la preparació de la comanda per al client. El fet d'haver de passar per N ubicacions és la base per a què aquest algorisme sigui recursiu. És a dir, aquest algorisme és recursiu perquè cal buscar entre tots els camins possibles de pas que hi han en el nostre magatzem el camí més òptim per a la recollida dels productes, tenint en compte, el cost ("pes") del camí

esgotant totes les possibles combinacions. El pes d'un camí es defineix com el total de distàncies que existeixen entre els diferents punts d'un camí.

La combinació de tots els possibles camins existents en un magatzem, es calcula mitjançant la combinació d'ubicacions per on s'han de passar a recollir els diferents productes per preparar la comanda i, tindrà com a resultat el factorial del nombre d'ubicacions (Número d'ubicacions!).

Deixant de banda, la recursivitat de l'algorisme passarem tot seguit a parlar de la base d'aquest algorisme. En primer lloc, i com en tot procés de Picking és necessari l'entrada d'una comanda de venta en el sistema perquè es produeixi el Picking i si cal la reposició de la ubicació de Picking. Per tant, el primer que ha de tenir en compte l'algorisme implementat és la comanda de venta amb els productes que s'han de servir. De la comanda de venta ens hem de quedar amb les referències (codi d'article) dels productes a servir, així com:

- La quantitat que s'ha d'entregar.

- El magatzem d' on s'ha de servir la mercaderia.

- La data d'entrega sol·licitada per part del client i, la data d'enviament que es té prevista perquè la mercaderia surti del magatzem i sigui enviada.

Un cop hem guardat les referències dels productes que cal servir, consultarem al mestre d'articles quines són les característiques que aquests presenten. Les consultes a realitzar són:

- Volum de l'article

- Pes que suporta l'article.

Cal fer aquestes dues consultes perquè el següent pas a seguir consisteix en ordenar aquests productes atenent al seu volum i al pes que han d'aguantar. A fi d'aconseguir, que la ruta òptima de recollida depengui entre d'altres coses de les característiques físiques dels productes a servir.

Una vegada tenim ordenats els productes de la comanda de venta, procedim a la seva preparació i a iniciar el procés de Picking / recollida de la mercaderia. En primer lloc, cal saber quin número de lot hem d'entregar al client en funció de la data de caducitat del producte. L'algorisme conté un procés que calcula quina és la data de caducitat del producte en funció del destí en el qual s'ha de servir el gènere al client i, ens retorna el número de lot del producte a servir. Per a la realització del càlcul s'estableix un marge de dies per a cada client tenint en compte la caducitat dels productes. Un cop tenim calculats el dies que tardarà el producte a caducar, (en funció del marge establert per al client i els dies que triga en arribar la mercaderia) el següent pas és triar un número de lot d'aquest producte. Per fer-ho s'utilitza o bé la valoració FIFO o LEFO segons ho estableixi el client. Per exemple: si el client té predefinit el mètode FIFO per a l'entrega dels productes, es servirà el número de lot que contingui la data de caducitat més propera als dies calculats prèviament. En canvi, si el client conté el mètode de valoració LEFO, es servirà el número de lot que contingui la data de caducitat més llunyana als dies calculats prèviament. En el cas, que no existeixi un número de lot a servir no es podrà realitzar la preparació del producte i, el sistema informarà que no es pot servir i que s'ha de realitzar una compra de l'article amb el número de lot que s'ha calculat.

Un cop sabem amb exactitud els productes que s'han de preparar, l'algorisme ha de comprovar si disposem de la quantitat suficient d'estoc en la ubicació de Picking de l'article per realitzar l'entrega al client amb el número de lot, prèviament calculat. Per saber en quina ubicació es troba localitzat el producte hem d'informar-la en la comanda de venta. Cal destacar, que cada article té una configuració prèvia en el sistema

on s'indica per magatzem quina és la ubicació del producte, de manera que, tenim configurat un magatzem amb ubicacions fixes per als articles que s'hi emmagatzemen. Per tant, ens serà relativament fàcil especificar en la comanda de venta on es troba emmagatzemant el producte; ja que, solament hem de realitzar una consulta en l'anomena't formulari d'articles per magatzem que conté Axapta. Aquest formulari és l'encarregat d'informar-nos on es troben ubicats els productes dins del magatzem.

Suposem que en la ubicació de Picking no hi ha la suficient quantitat de producte en estoc necessària per preparar l'entrega al client. Aleshores realitzarem una consulta en la ubicació d'emmagatzemament del producte. En el cas, que en la ubicació de Picking no hi hagi la quantitat suficient del producte però en la ubicació d'emmagatzemament si que hi sigui, el sistema mitjançant un informe informarà(a partir d'ara informe de Picking i s'explicarà més endavant) que s'ha d'aprovisionar[7] la ubicació de Picking. En cas contrari, si no disposem d'estoc en la ubicació de Picking ni en la ubicació d'emmagatzemament, el sistema informarà en l'informe de Picking, que no s'ha de realitzar cap operació per a la preparació de la comanda d'aquell article. El procés de Picking i de reposició implementat en aquest projecte va guardant les operacions que s'han de realitzar. Això, permet que l'usuari pugui veure mitjançant l'informe de Picking si s'ha pogut servir tota la comanda al client o no. Les operacions de les quals estem parlant són les següents:

- Picking: quan s'ha realitzat correctament la recollida del producte per a la ubicació de Picking.

- Reposició / aprovisionament: quan en la ubicació de Picking no hi ha prou estoc a servir i s'ha d'aprovisionar de la ubicació d'emmagatzemament.

[7]Aprovisionar, omplir la ubicació de Picking mitjançant els productes de la ubicació d'emmagatzemament.

- No operació: quan no es pot servir el producte per falta d'estoc al magatzem, tant en la ubicació de Picking com en la de reposició.

Per servir la comanda al client s'haurà de realitzar una compra del producte sol·licitat. Si en la ubicació de Picking disposem de la quantitat del producte per poder complir amb el servei del client, el sistema és guardarà que ha de realitzar l'operació de Picking i continuarà amb el següent procés. Aquest procés forma part de la segona fase de l'algorisme i, consisteix en el càlcul de la ruta òptima per a la recollida dels productes tenint en compte l' anteriorment esmentat.

En aquesta segona fase de l'algorisme, en primer lloc, cal consultar quines són les operacions que s'han de realitzar. El procés guarda en una taula, anomenada Operacions, totes les tasques a realitzar; així com, les ubicacions que l'usuari del magatzem ha de recórrer. Sabent quines són les ubicacions que cal "visitar" l'algorisme calcula tots els possibles camins de recollida dels productes. Aquests possibles camins són el resultat de permutar les N ubicacions que l'usuari ha de recórrer dins del magatzem per realitzar el compliment de la comanda de venta. De tots els possibles camins existents s'ha de tenir en compte segons les característiques dels productes quin serà el millor per a la seva recollida. És a dir, no podem escollir qualsevol camí perquè segons com recollim els productes pot succeir que els que estiguin situats a sota es trenquin o es facin malbé. Per tant, per escollir el camí ideal cal tenir en compte el pes del producte i el pes màxim que aquest aguanta.

Finalment, un cop calculats els camins els quals s'han obtingut tenint en compte les característiques dels productes. L'algorisme ha de calcular els pesos dels camins per a l'obtenció de la ruta de recollida més òptima. Els pesos dels camins van en funció del passadís, prestatge i balda. Per exemple: si ens trobem situats en el passadís número 1 del magatzem i hem d'anar al passadís número 2, el sistema sumarà la diferència entre els números de passadissos i, en aquest cas, el pes a

sumar per passadís seria 1. El mateix succeiria per al càlcul dels prestatges. Veiem una sèrie d'exemple per a entendre-ho millor:

- Ubicació 1: 01-1-01-1

- Ubicació 2: 03-2-04-1

    - Pes del camí per anar de la ubicació 1 a la 2: |(1-3)| + |(1-2)| + |(1-4)| = 6

- Ubicació 3: 02-3-01-3

- Ubicació 4: 01-4-02-1

    - Pes del camí per anar de la ubicació 3 a la 4: |(2-1)| + |(3-4)| + |(1-2)| = 3

5.2. Anàlisis dels magatzems

Quantes vegades ens hem preguntat quina seria la millor manera de distribuir els productes en un magatzem? Aquesta és la pregunta que intentarem resoldre en aquest capítol.

A més, en aquest capítol podrem veure i analitzar els dissenys de magatzems que utilitzarem per realitzar l'algorisme. És important tenir una bona estructura de magatzem, ja que, això permet a l'algorisme funcionar molt més ràpid alhora d'escollir el material per complir amb la demanda del client.

Primerament, caldrà realitzar un anàlisis detallat de les característiques físiques que presenten els productes abans d'entrar amb els requeriments tècnics i d'organització del magatzem.

Cal destacar els següents factors envers les característiques dels productes:

- Característiques físiques
    - Volum
    - Pes
- Caducitat
    - Productes de llarga duració
    - Productes de caducitat fixa

Per altra banda, cal analitzar la distribució dels productes al magatzem. Podem distribuir el magatzem en diferents zones en funció de la rotació dels productes. Aquesta rotació vindrà determinada per la quantitat de ventes que es facin de cada producte. Finalment, tindrem distribuït el magatzem en tres zones diferents:

- Zona A: zona situada el més a prop possible del moll de sortida. Hi col·locarem els productes de més rotació, és a dir, aquells que es venen més.

- Zona B: Hi hauran els productes del tipus B a una distància més o menys a prop de la sortida.

- Zona C: Hi hauran els productes del tipus C, aquests al tenir molt menys rotació que els anteriors els podem situar més lluny de la zona de sortida.

Utilitzarem una estructura genèrica per a la distribució dels productes en les dues estructures de magatzem. D'aquesta forma, utilitzant la mateixa redistribució del material, segons una sèrie de regles, podrem provar en

funció dels dos dissenys que estudiarem quin és el més adequat i eficient alhora de provar l'algorisme.

D'altra banda, el magatzem estarà composat per dues zones una àrea / moll d'entrada i una altra de sortida. En l'àrea de sortida es prepararan les comandes per servir als client. A més a més, el magatzem estarà composat per quatre passadissos. Aquests passadissos tindran dos prestatgeries en cadascun d'ells. Cada prestatgeria contindrà quatre baldes amb les seves respectives posicions per a la col·locació dels productes.

Les regles que s'utilitzaran per a la col·locació dels productes, en funció de les seves característiques físiques, són les següents:

- Els productes més pesants i els més voluminosos es situaran a les baldes més baixes de la prestatgeria. En canvi, els productes més lleugers es col·locaran a les zones més altes. D'aquesta manera, no caldrà fer tan esforç alhora de manipular-los.

- Dividirem la prestatgeria en dues parts molt importants. En una d'elles hi col·locarem els productes per al Picking i en l'altra els productes per a la reposició. La manera de distribuir-los serà la següent:

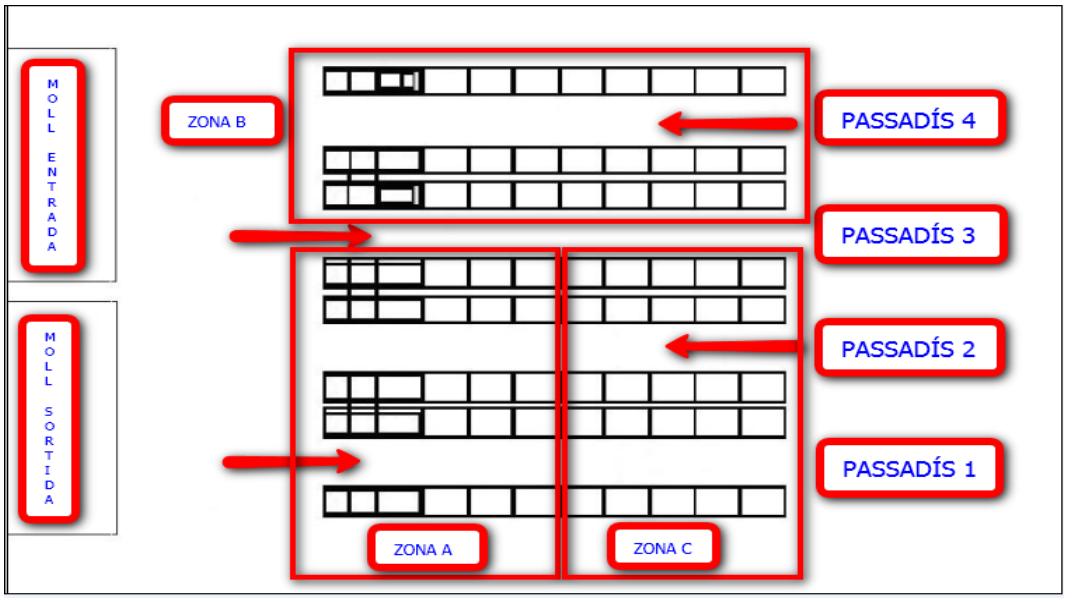
    - Cada prestatgeria conté quatre baldes. Utilitzarem la primera balda començant per baix per als productes per a Picking, la següent per als productes de reposició i així consecutivament.

      Per tant, tots els prestatges del magatzem contindran posicions per a Picking i per a la reposició dels productes. En un dels magatzems podem utilitzar aquesta forma de reestructuració.

- Es col·locaran al davant del prestatge els productes que caduquen més aviat i s'han de servir en primer lloc.

5.2.1. Estructura del primer magatzem

El primer disseny que estudiarem és el següent:

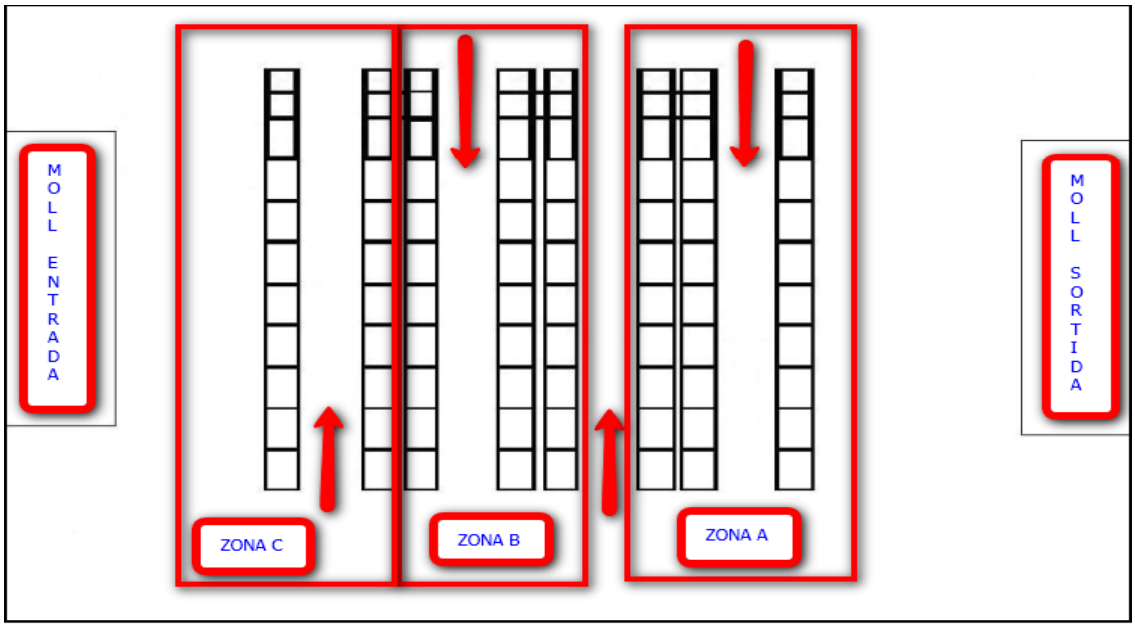


Il·lustració 14 Estructura del primer magatzem

Com podem veure en la imatge anterior s'ha estructurat el magatzem de forma que el moll d'entrada és al costat del moll de sortida. S'ha dividit el magatzem en tres zones diferents en la zona A, hi col·locarem els productes que més ingressos generen en la zona C, els que menys ingressos aporten i la resta de productes els col·locarem en la zona B.

5.2.2. Estructura del segon magatzem

El segon disseny que estudiarem és el següent:



Il·lustració 15 Estructura del segon magatzem

En aquest disseny tenim el moll d'entrada en un extrem del magatzem i el moll de sortida en l'altre. Les zones estan distribuïdes de forma diferent a les del primer disseny de magatzem. D'aquesta manera, podrem esbrinar mitjançant l'algorisme quin dels dos dissenys de magatzem és més eficient alhora d'iniciar el Picking. En cadascuna de les zones es farà servir la mateixa distribució dels productes utilitzada en el primer disseny de magatzem, o sigui, els productes que es venen més els ubicarem a la zona A, els que tenen menys rotació a la zona C i la resta a la zona B. A més a més, tindrem en compte les distàncies entre els molls d'entrada i de sortida per al càlcul de la ruta òptima.

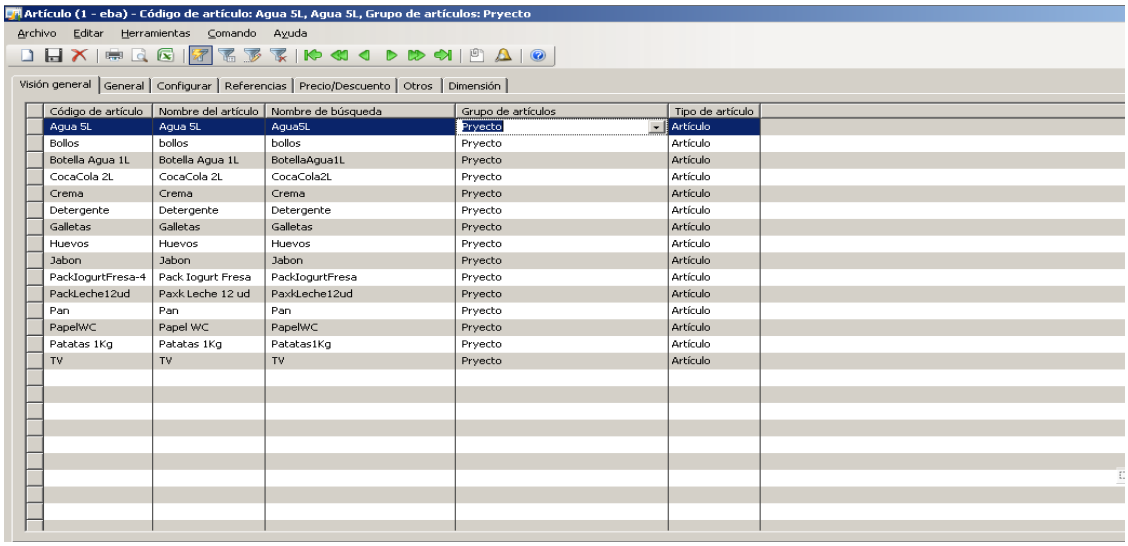
5.3. Configuració en AX per a la implementació de l'algorisme

En aquest apartat explicarem quines són les configuracions necessàries que s'han de realitzar en l'aplicació, per a la implementació i la realització de l'algorisme.

5.3.1. Articles

Per a la realització d'aquest algorisme tindrem en compte una sèrie d'articles que s'hauran de recollir en el procés de Picking. Aquests articles fan referència a la cadena alimentària.

Els articles que s'utilitzen en aquest projecte són els següents:



Código de artículo	Nombre del artículo	Nombre de búsqueda	Grupo de artículos	Tipo de artículo
Agua 5L	Agua 5L	Agua5L	Pryecto	Artículo
Bollos	bollos	bollos	Pryecto	Artículo
Botella Agua 1L	Botella Agua 1L	BotellaAgua1L	Pryecto	Artículo
CocaCola 2L	CocaCola 2L	CocaCola2L	Pryecto	Artículo
Crema	Crema	Crema	Pryecto	Artículo
Detergente	Detergente	Detergente	Pryecto	Artículo
Galletas	Galletas	Galletas	Pryecto	Artículo
Huevos	Huevos	Huevos	Pryecto	Artículo
Jabon	Jabon	Jabon	Pryecto	Artículo
PackIogurtFresa-4	Pack Iogurt Fresa	PackIogurtFresa	Pryecto	Artículo
PackLeche12ud	Paxk Leche 12 ud	PaxkLeche12ud	Pryecto	Artículo
Pan	Pan	Pan	Pryecto	Artículo
PapelWC	Papel WC	PapelWC	Pryecto	Artículo
Patatas 1Kg	Patatas 1Kg	Patatas1Kg	Pryecto	Artículo
TV	TV	TV	Pryecto	Artículo

Il·lustració 16 Articles del projecte

Tots aquets articles tindran assignat el mateix grup de dimensions. Però no tots ells comparteixen les mateixes característiques de pes, volum, pes de resistència màxima, unitat de venta, etc.

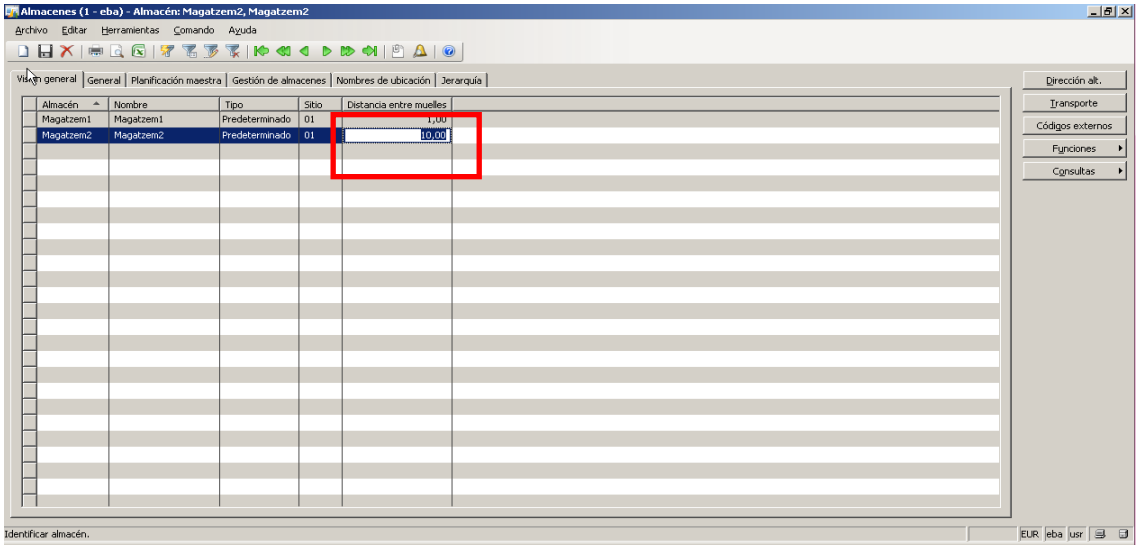
De cadascun dels productes creats veurem tot seguit quines són les seves característiques físiques (pes, volum i pes màxim de resistència):

Article	Volum	Pes	Pes aguant màxim
Agua 5L	5	5	3,5
Bollos	5	0,3	0,5
Botella Agua 1L	1	1	3
Coca Cola 2L	2	2	3
Crema	0,5	0,4	0,6
Detergente	6	5	4
Galletas	0,7	0,3	0,5
Huevos	0,2	0,1	0,1
Jabon	1,2	1	2,5
Pack Iogurt 4 ud	0,4	0,2	0,5
Pack Leche 12 ud	5	12	5
Pan	1	0,5	0,5
Papel WC	5	0,05	0,75
Patatas 1Kg	8	1	3
TV	20	60	20

5.3.2. Magatzems

El formulari de Magatzems està format per diverses pestanyes i botons que ens permeten diferents funcionalitats.

La fitxa Visió General ens permet crear un magatzem nou, indicar-li el tipus com per exemple: si es tracta d'un magatzem de quarantena i, permet si es el cas assignar un lloc al magatzem ("sitio": nom en Axapta). A més a més, informarem de la distància que hi ha entre el moll d'entrada i el moll de sortida. Si ens fixem en la imatge següent podem veure que la distància entre els molls del primer magatzem és d'1 metre en canvi la distància per al segon magatzem és de 10 metres.



Il·lustració 17 Pestanya Visió General d'un magatzem

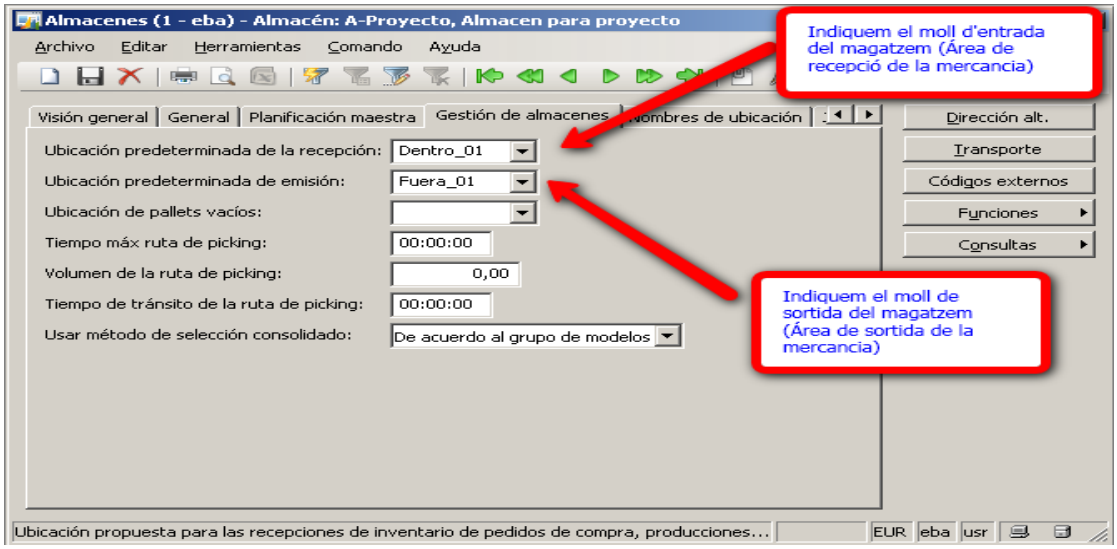
La fitxa General ens indica el tipus de magatzem. A més a més,en el cas que tinguem un magatzem predeterminat tenim la possibilitat d'assignar-li un magatzem de quarantena o bé un magatzem de trànsit.

D'altra banda, també podem indicar-li si el magatzem està ubicat en un proveïdor.

La fitxa anomenada planificació d'articles ens permet configurar el calendari de treball d'un magatzem. Aquest calendari s'utilitza per calcular la data de la comanda, data de requisit i la data d'entrega preferent.

Per configurar el nostre magatzem nosaltres no utilitzarem la fitxa General ni la de la planificació.

Una altra part que configura el formulari de magatzems és la pestanya anomenada Gestió de magatzems. Aquesta pestanya ens permet definir quina és la zona d'entrada i de sortida de les mercaderies d'un magatzem.



Il·lustració 18 Pestanya Gestió de magatzems del formulari magatzem

5.3.3. Col·locació dels productes en les ubicacions del magatzem

En Axapta disposem de dues maneres per realitzar la col·locació dels productes en les diferents ubicacions del magatzem. Aquests processos que explicarem a continuació, s'anomenen:

- Articles per magatzem

- Recepció d'articles

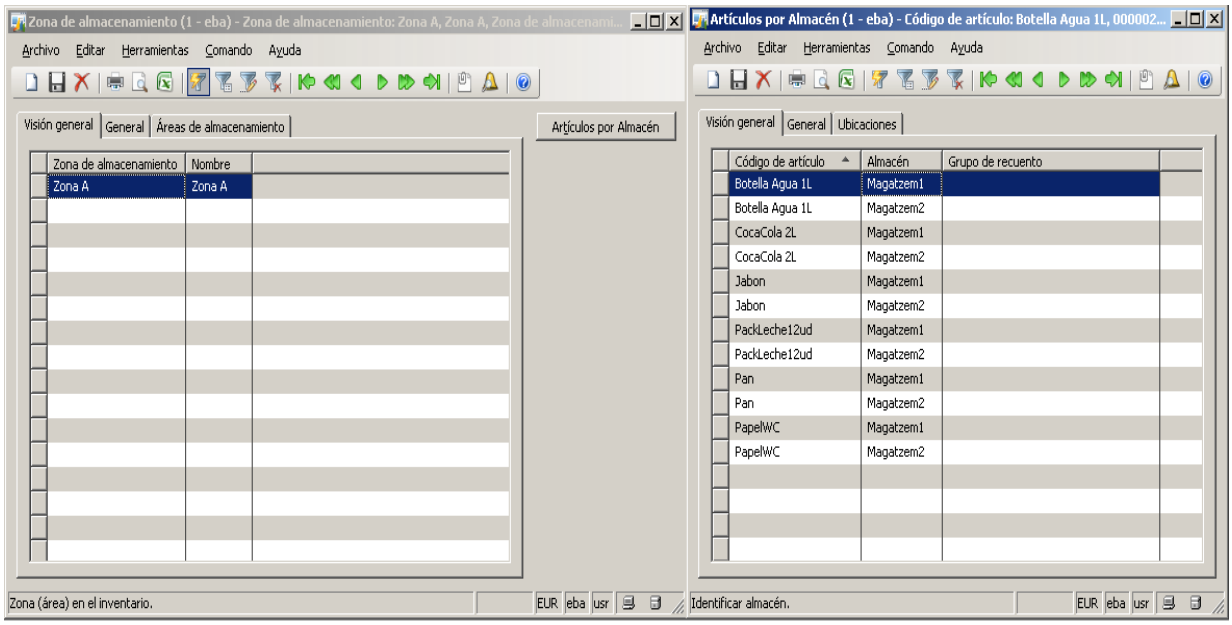
Per realitzar el projecte utilitzarem el procediment d'articles per magatzem i considerarem que s'han realitzat les compres oportunes per disposar de suficient estoc de tots els productes.

5.3.3.1. ARTICLES PER MAGATZEM

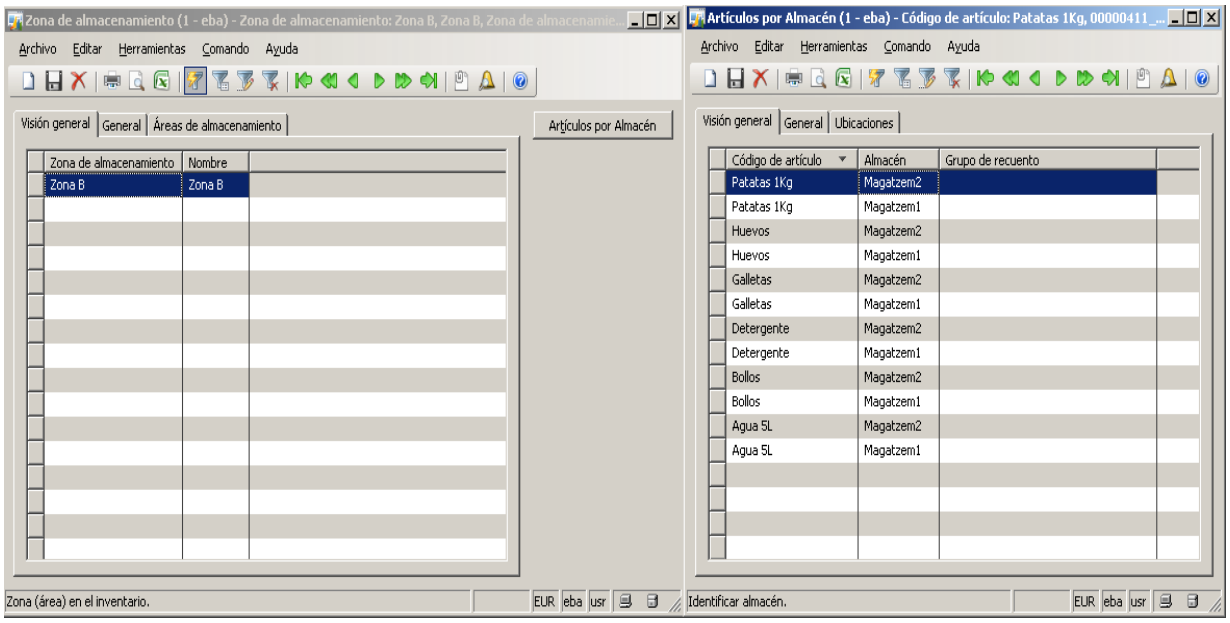
Una vegada hem configurat l'estructura del magatzem ja podem assignar els articles a les seves ubicacions de Picking i de Reposició corresponents, de tal forma que, tindrem una sèrie d'articles per a cada zona del magatzem. Mitjançant el botó que tenim a la imatge anterior anomenat "Articles per magatzem", assignarem els articles a cada zona.

Per exemple: en la zona A tenim assignats els següents productes, ja que, els podem considerar com els que més beneficis generen a la empresa perquè són aquells que més es venen.

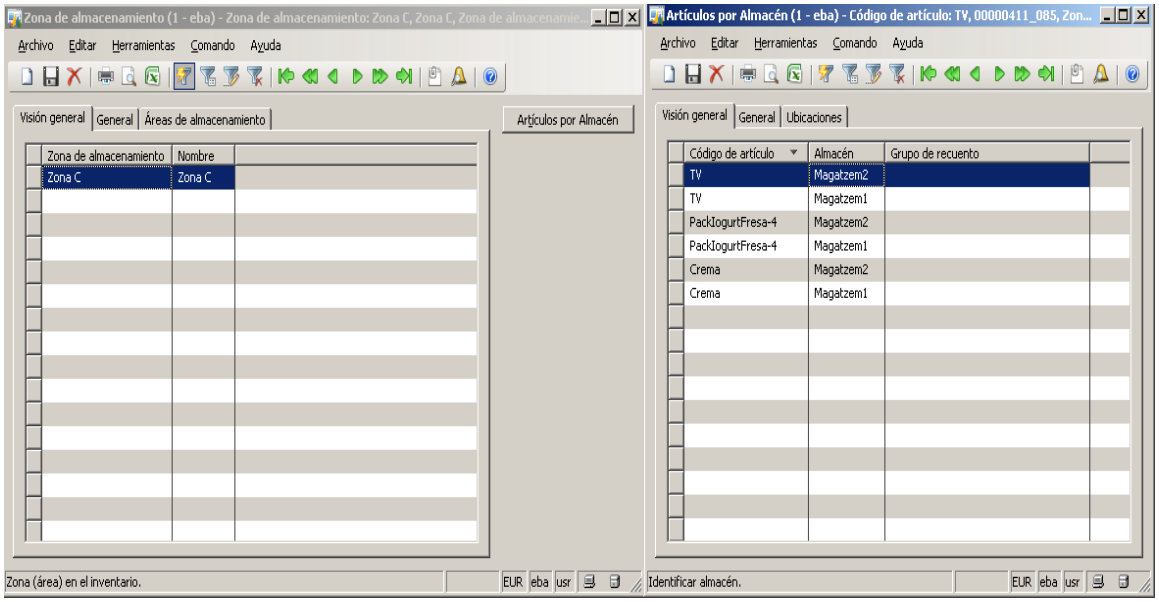
Podem veure els articles que estan assignats en les diferents zones del magatzem en la configuració d' Axapta:



Il·lustració 19 Articles per magatzem Zona A



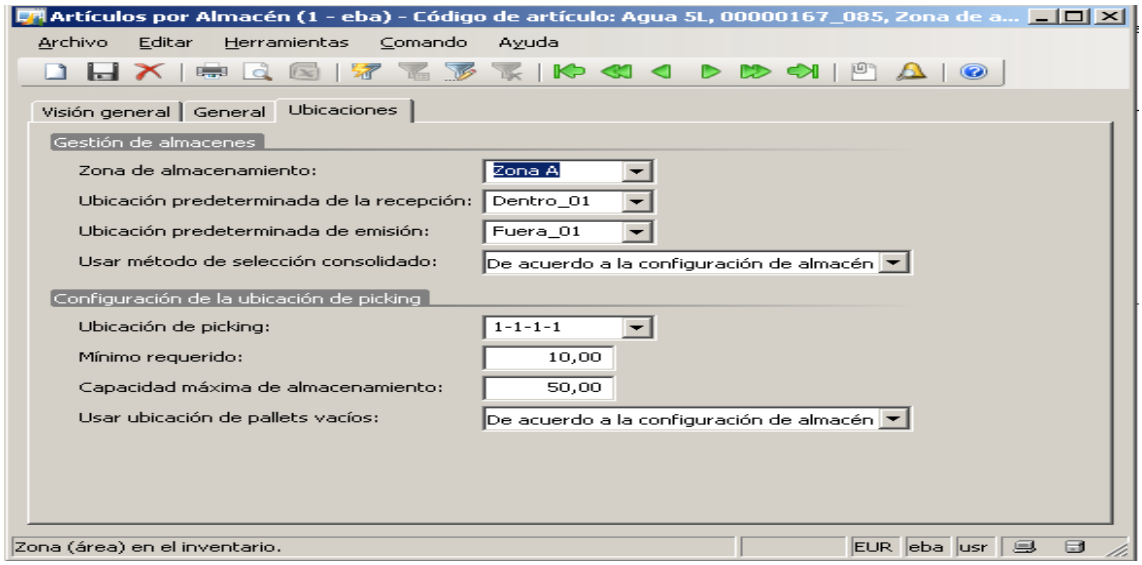
Il·lustració 20 Articles per magatzem Zona B



Il·lustració 21 Articles per magatzem Zona C

La fitxa General del mateix formulari conté la mateixa informació que la fitxa de visió general.

La fitxa d'ubicacions ens permet indicar quina serà la ubicació de cada article dins del magatzem. D'aquesta manera quan iniciem el Picking haurem d'anar a buscar a la ubicació que té assignada cada producte.



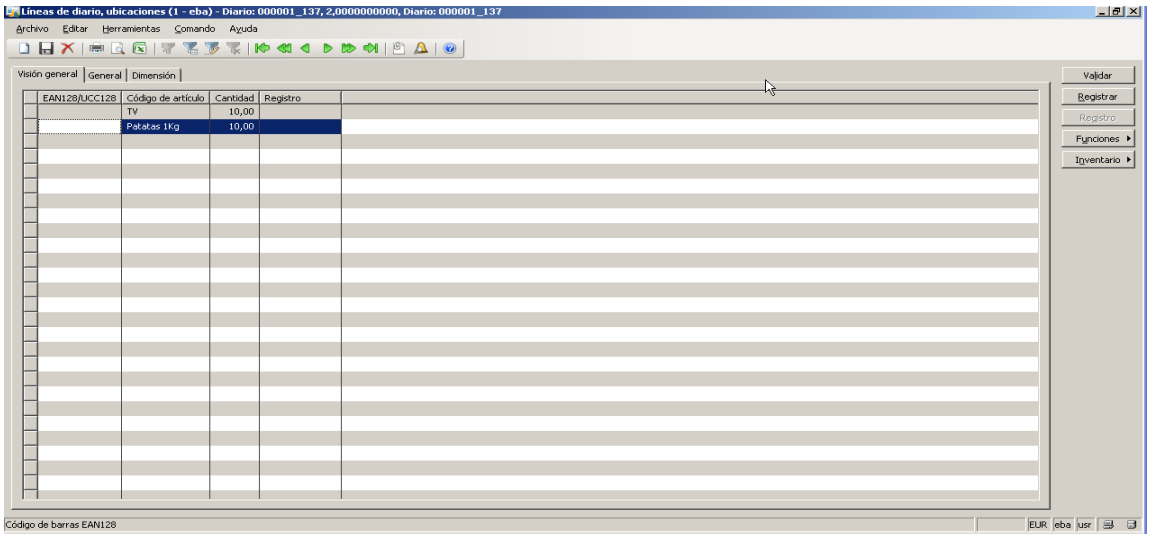
Il·lustració 22 Pestanya Ubicacions del formulari Articles per magatzem

Com podem veure en el formulari anterior a cada article podem indicar-li la seva zona d'emmagatzemament, el seu moll d'entrada i de sortida, la seva ubicació de Picking i la quantitat mínima i màxima de productes que hi ha d'haver emmagatzemada en la ubicació de Picking.

5.3.3.2. Recepció d'articles

En el cas de tenir un magatzem amb ubicacions fixes, com és el cas dels dos dissenys implementats en aquest projecte, l'entrada de productes al magatzem es fa mitjançant el concepte de diari. Aquest concepte ens permet entrar producte a producte indicant la quantitat, el magatzem i la ubicació on volem que s'emmagatzemi.

El diari per a l'entrada dels productes en les diferents ubicacions del magatzem s'anomena "Recepció d'articles". En la primera pestanya anomenada "Visió General" indiquem el producte que volem rebre i la quantitat.



Il·lustració 23 Pestanya Visió General de la recepció d'articles

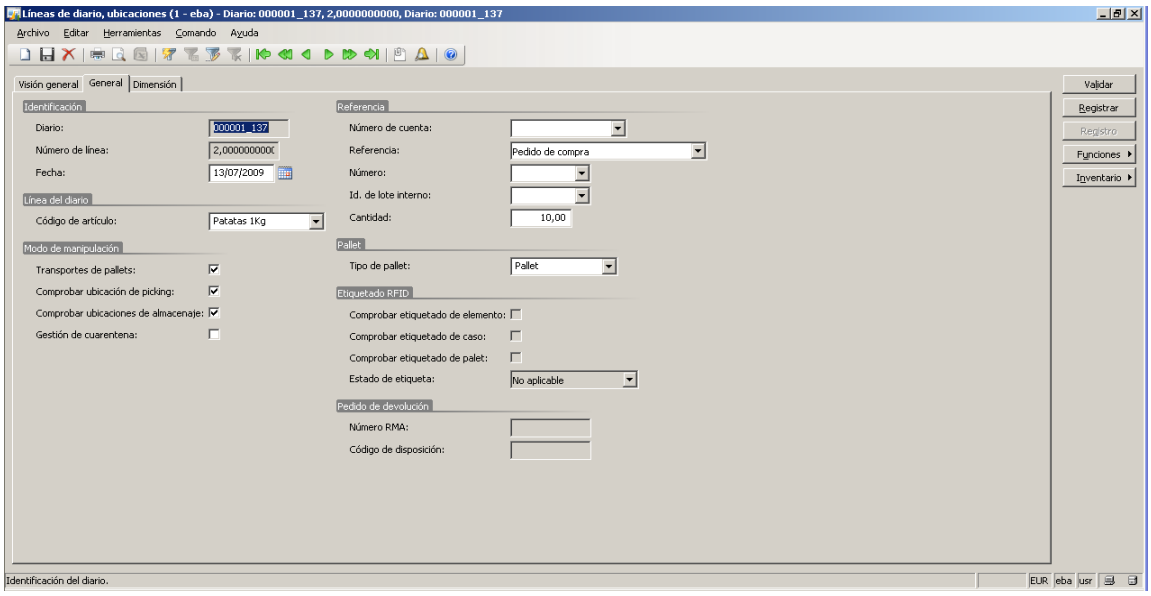
En aquesta segona pestanya anomenada, "General" cal destacar tres punts molt importants que són:

- Transport de pallets: si el marquem voldrà dir que els articles enregistrats en el diari es mouran dins del magatzem mitjançant el

transport de pallets. Cal especificar una ubicació del tipus Moll en aquest cas moll d'entrada i, a més a més, utilitzar un identificador de pallet (habilitar una dimensió d'inventari) per a què el diari es registri de forma correcta.

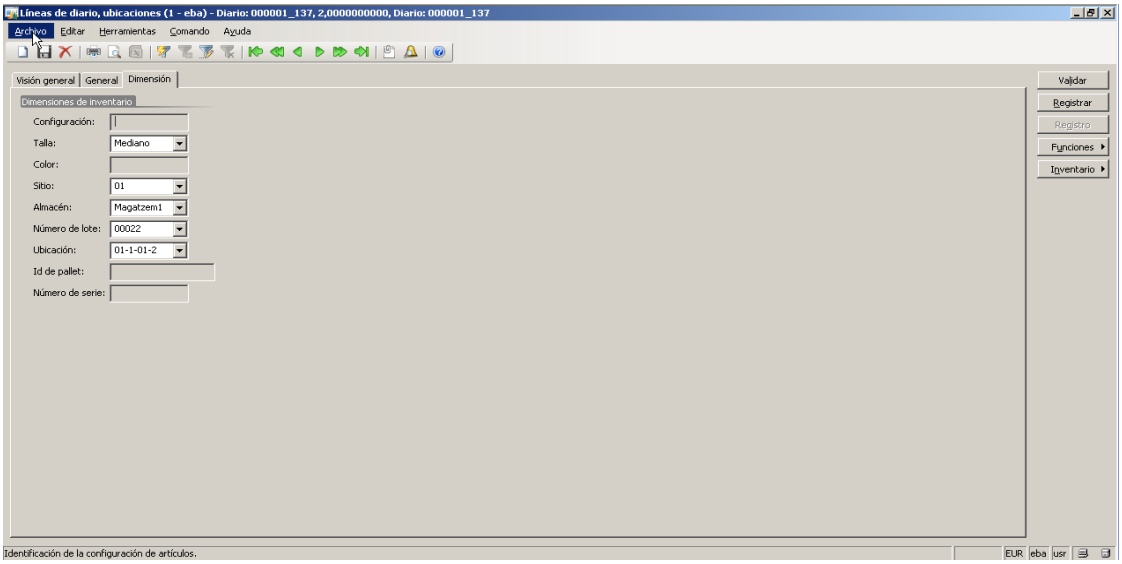
- Comprovar ubicació de Picking: el marcarem quan vulguem que Microsoft Dynamics AX, comprovi de forma automàtica l'espai que hi ha disponible per a l'article en la seva ubicació de Picking habitual.

- Comprovar ubicacions d'emmagatzemament: cal especificar si desitgem que es trobi una ubicació d'emmagatzemament per a l'article en la zona d'emmagatzemament del mateix.



Il·lustració 24 Pestanya General de la recepció d'articles

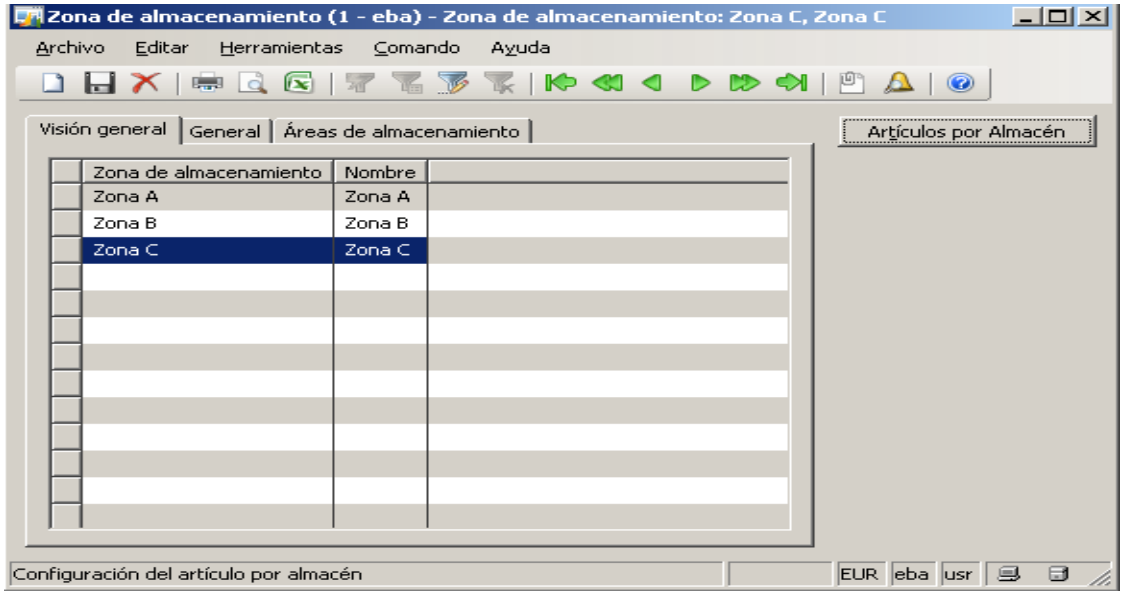
En aquesta pestanya anomenada "Dimensió", s'especifica on volem guardar el producte. És a dir, ens informa del lloc, del magatzem, de la ubicació on el guardarem i, a més a més, ens indica el número de lot del producte.



Il·lustració 25 Pestanya Dimensions de la recepció d'articles

5.3.4. ZONES

Podem dividir el magatzem en diferents zones. En la zona A, hi col·locarem els productes que més es venen. En la zona C, els productes que menys sortida tenen i en la zona B la resta dels articles.

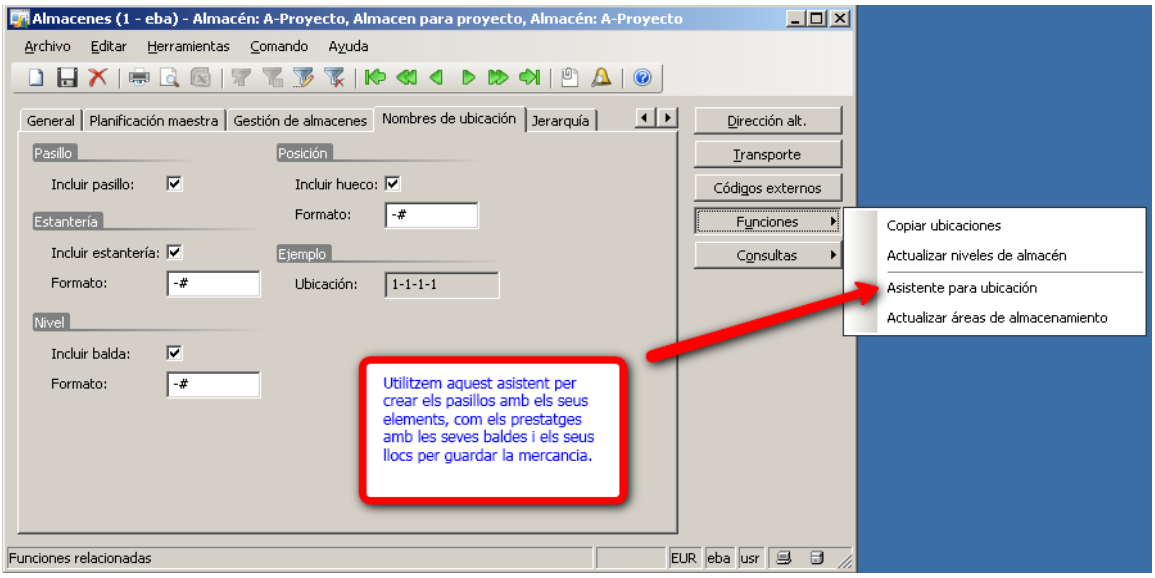


Il·lustració 26 Zones

5.3.5. UBICACIONS

Des del formulari de magatzems tenim l'opció de generar les ubicacions del magatzem.

La fitxa Noms d'ubicació, ens permet generar el format de la ubicació del nostre magatzem tenint en compte si s'utilitzen els passadissos, les prestatgeries, els nivells (anomenats "baldes" en AX) i les posicions (anomenades "huecos" en AX). També ens permet indicar un format per als diferents elements dels que es composa un magatzem.



Il·lustració 27 Pestanya Noms d'Ubicacions del formulari magatzem

Finalment, tenim la fitxa anomenada Jerarquia. Aquesta fitxa ens permet veure l'estructura dels llocs creats en Axapta amb els seus diferents magatzems. És a dir, podem tenir creats diferents llocs, com per exemple: Lleida, Barcelona o Girona, etc. I per a cadascun d'aquests llocs definir i assignar els magatzems que hi formen part.

Mitjançant el botó consulta d'aquest formulari podem veure quines són les seves ubicacions i els seus passadissos.

En aquest podem veure les diferents ubicacions que s'han creat a partir de l'assistent d'ubicacions d'Axapta[8]. A més a més, podem observar

quina és la distància de cada ubicació respecte al moll de sortida. Aquests dos magatzems estan distribuïts de forma diferent, la qual cosa, implica que les distàncies envers el moll de sortida siguin diferents per a cadascuna de les ubicacions existents, malgrat que, aquestes s'anomenin igual.

Ubicaciones (1 - eba) - Ubicación: Ent_01, Magatzem1, Almacén: Magatzem1

Archivo Editar Herramientas Comando Ayuda

Visión general | General | Otros | Estructura en árbol

Almacén	Ubicación	Tipo de ubicación	Nº máx. de pallets	Nº de pallets ubicados	Espacio disponible, en pallets	Distancia entre muelles
Magatzem1	01-1-01-1	Ubicación de picking	1	0	1	1,00
Magatzem1	01-1-01-2	Ubicación de picking	1	0	1	2,00
Magatzem1	01-1-02-1	Ubicación de almacenaje	1	0	1	3,00
Magatzem1	01-1-02-2	Ubicación de almacenaje	1	0	1	4,00
Magatzem1	01-1-03-1	Ubicación de picking	1	0	1	5,00
Magatzem1	01-1-03-2	Ubicación de picking	1	0	1	6,00
Magatzem1	01-1-04-1	Ubicación de almacenaje	1	0	1	7,00
Magatzem1	01-1-04-2	Ubicación de almacenaje	1	0	1	8,00
Magatzem1	01-2-01-1	Ubicación de picking	1	0	1	1,00
Magatzem1	01-2-01-2	Ubicación de picking	1	0	1	2,00
Magatzem1	01-2-02-1	Ubicación de almacenaje	1	0	1	3,00
Magatzem1	01-2-02-2	Ubicación de almacenaje	1	0	1	4,00
Magatzem1	01-2-03-1	Ubicación de picking	1	0	1	5,00
Magatzem1	01-2-03-2	Ubicación de picking	1	0	1	6,00
Magatzem1	01-2-04-1	Ubicación de almacenaje	1	0	1	7,00
Magatzem1	01-2-04-2	Ubicación de almacenaje	1	0	1	8,00
Magatzem1	02-1-01-1	Ubicación de picking	1	0	1	1,00
Magatzem1	02-1-01-2	Ubicación de picking	1	0	1	2,00
Magatzem1	02-1-02-1	Ubicación de almacenaje	1	0	1	3,00
Magatzem1	02-1-02-2	Ubicación de almacenaje	1	0	1	4,00
Magatzem1	02-1-03-1	Ubicación de picking	1	0	1	5,00
Magatzem1	02-1-03-2	Ubicación de picking	1	0	1	6,00
Magatzem1	02-1-04-1	Ubicación de almacenaje	1	0	1	7,00
Magatzem1	02-1-04-2	Ubicación de almacenaje	1	0	1	8,00
Magatzem1	02-2-01-1	Ubicación de picking	1	0	1	1,00

Consultas | Transportes de pallets | Funciones | Imprimir

Ubicaciones que no están reservadas ni ocupadas, en pallets.

EUR eba usr

Il·lustració 28 Ubicacions primer magatzem

Ubicaciones (1 - eba) - Ubicación: Sal_01, Magatzem2, Almacén: Magatzem2

Archivo Editar Herramientas Comando Ayuda

Visión general | General | Otros | Estructura en árbol

Almacén	Ubicación	Tipo de ubicación	Nº máx. de pallets	Nº de pallets ubicados	Espacio disponible, en pallet	Distancia entre muelles
Magatzem2	01-1-01-1	Ubicación de picking	1	0	1	1,00
Magatzem2	01-1-01-2	Ubicación de picking	1	0	1	2,00
Magatzem2	01-1-02-1	Ubicación de almacenaje	1	0	1	3,00
Magatzem2	01-1-02-2	Ubicación de almacenaje	1	0	1	4,00
Magatzem2	01-1-03-1	Ubicación de picking	1	0	1	5,00
Magatzem2	01-1-03-2	Ubicación de picking	1	0	1	6,00
Magatzem2	01-1-04-1	Ubicación de almacenaje	1	0	1	7,00
Magatzem2	01-1-04-2	Ubicación de almacenaje	1	0	1	8,00
Magatzem2	01-2-01-1	Ubicación de picking	1	0	1	8,00
Magatzem2	01-2-01-2	Ubicación de picking	1	0	1	7,00
Magatzem2	01-2-02-1	Ubicación de almacenaje	1	0	1	6,00
Magatzem2	01-2-02-2	Ubicación de almacenaje	1	0	1	5,00
Magatzem2	01-2-03-1	Ubicación de picking	1	0	1	4,00
Magatzem2	01-2-03-2	Ubicación de picking	1	0	1	3,00
Magatzem2	01-2-04-1	Ubicación de almacenaje	1	0	1	2,00
Magatzem2	01-2-04-2	Ubicación de almacenaje	1	0	1	1,00
Magatzem2	02-1-01-1	Ubicación de picking	1	0	1	10,00
Magatzem2	02-1-01-2	Ubicación de picking	1	0	1	11,00
Magatzem2	02-1-02-1	Ubicación de almacenaje	1	0	1	12,00
Magatzem2	02-1-02-2	Ubicación de almacenaje	1	0	1	13,00
Magatzem2	02-1-03-1	Ubicación de picking	1	0	1	14,00
Magatzem2	02-1-03-2	Ubicación de picking	1	0	1	15,00
Magatzem2	02-1-04-1	Ubicación de almacenaje	1	0	1	16,00
Magatzem2	02-1-04-2	Ubicación de almacenaje	1	0	1	10,00
Magatzem2	02-2-01-1	Ubicación de picking	1	0	1	17,00

Consultas | Transportes de pallets | Funciones | Imprimir

Identificar almacén.

EUR eba usr

Il·lustració 29 Ubicacions segon magatzem

[8] Assistent d'ubicacions d'Axapta, procés que utilitza el sistema per a la creació d'ubicacions d'un magatzem, així com, la creació dels molls d'entrada i de sortida, els passadissos , les prestatgeries i les baldes de cadascun dels prestatges.

Segons l'estructura del primer magatzem es realitza la distribució de les ubicacions en les diferents zones de forma lògica:

ZONA A	ZONA B	ZONA C
- 01-1-01-1	- 03-2-01-1	- 01-1-03-1
- 01-1-01-2	- 03-2-01-2	- 01-1-03-2
- 01-1-02-1	- 03-2-02-2	- 01-1-04-1
- 01-1-02-2	- 03-2-03-1	- 01-1-04-2
- 01-2-01-1	- 03-2-03-2	- 01-2-03-1
- 01-2-01-2	- 03-2-04-1	- 01-2-03-2
- 01-2-02-1	- 03-2-04-2	- 01-2-04-1
- 01-2-02-2	- 04-1-01-1	- 01-2-04-2
- 02-1-01-1	- 04-1-01-2	- 01-2-04-2
- 02-1-01-2	- 04-1-02-1	- 02-1-03-1
- 02-1-02-1	- 04-1-02-2	- 02-1-03-2
- 02-1-02-2	- 04-1-03-1	- 02-1-04-1
- 02-2-01-1	- 04-1-03-2	- 02-1-04-2
- 02-2-01-2	- 04-1-03-2	- 02-2-03-1
- 02-2-02-1	- 04-1-04-1	- 02-2-03-2
- 02-2-02-2	- 04-1-04-2	- 02-2-04-1
- 03-1-01-1	- 04-2-01-1	- 02-2-04-2
- 03-1-01-2	- 04-2-01-2	- 03-1-03-1

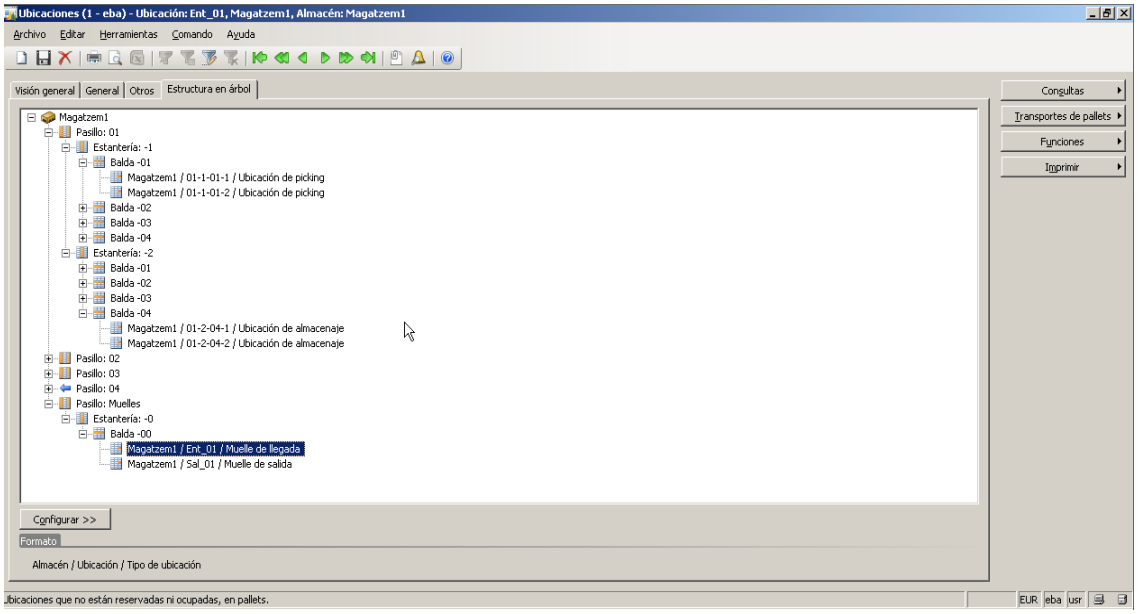
- 03-1-02-1 - 03-1-02-2	- 04-2-02-1 - 04-2-02-1 - 04-2-02-2 - 04-2-03-1 - 04-2-03-2 - 04-2-04-1 - 04-2-04-2	- 03-1-03-2 - 03-1-04-1 - 03-1-04-2

Veiem també la distribució de les ubicacions per al segon magatzem:

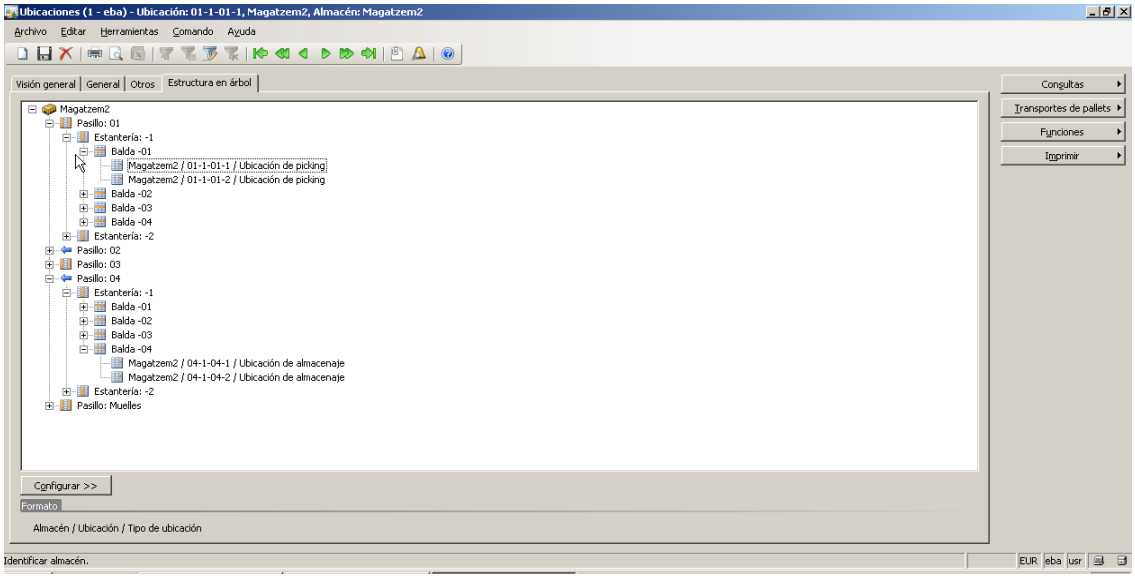
ZONA A	ZONA B	ZONA C
- 01-1-01-1	- 02-2-01-1	- 04-1-01-1
- 01-1-01-2	- 02-2-01-2	- 04-1-01-2
- 01-1-02-1	- 02-2-02-1	- 04-1-02-1
- 01-1-02-2	- 02-2-02-2	- 04-1-02-2
- 01-1-03-1	- 02-2-03-1	- 04-1-03-1
- 01-1-03-2	- 02-2-03-2	- 04-1-03-2
- 01-1-04-1	- 02-2-04-1	- 04-1-03-2
- 01-1-04-2	- 02-2-04-2	- 04-1-04-1
- 01-2-01-1	- 03-1-01-1	- 04-1-04-2
- 01-2-01-2	- 03-1-01-2	- 04-2-01-1
- 01-2-02-1	- 03-1-02-1	- 04-2-01-2
- 01-2-02-2	- 03-1-02-2	- 04-2-02-1
- 01-2-03-1	- 03-1-03-1	- 04-2-02-1
- 01-2-03-2	- 03-1-03-2	- 04-2-02-2
- 01-2-04-1	- 03-1-04-1	- 04-2-03-1
- 01-2-04-2	- 03-1-04-2	- 04-2-03-2
- 02-1-01-1	- 03-2-01-1	- 04-2-04-1
- 02-1-01-2	- 03-2-01-2	- 04-2-04-2
- 02-1-02-1	- 03-2-02-2	

- 02-1-02-2 - 02-1-03-1 - 02-1-03-2 - 02-1-04-1 - 02-1-04-2	- 03-2-03-1 - 03-2-03-2 - 03-2-04-1 - 03-2-04-2	

Podem veure també com quedaria l'estructura del magatzem mitjançant un esquema.



Il·lustració 30 Pestanya Estructura en Arbre de les ubicacions del magatzem 1



Il·lustració 31 Pestanya Estructura en Arbre de les ubicacions del magatzem 2

Tot i que Axapta mostra les mateixes estructures de magatzem. Cal dir que aquestes estructures són diferents, ja que, la distància existent entre les ubicacions i el moll de sortida; així com, la distància del moll de sortida respecte al moll d'entrada no són les mateixes. Precisament aquests són els paràmetres claus que s'han de tenir en compte alhora d'escollir el millor camí per a la recollida dels productes que s'han de servir al client.

5.3.6. Grup de dimensions d'inventari

Respecte als articles que s'utilitzen per realitzar aquest projecte, cal dir, que és necessari configurar-los per a què continguin una sèrie de dimensions, com són:

Dimensiones de artículo - activación y configuración

Visión general | General | Editar | Asignar un nuevo nombre

Activo	Nombre	Obligatorio	Existencias principales	Recepción en blanco ...	Emisión en blanco pe...	Inventario físico	Inventario financiero	Plan de cobertura por dimensión
☐	Configuración	☑	☑	☐	☐	☑	☑	☑
☐	Composicion	☑	☑	☐	☐	☑	☑	☑
☐	Acabado	☑	☑	☐	☐	☑	☑	☑

Dimensiones de almacenamiento - activación y configuración

Visión general | General | Editar

Activo	Nombre	Obligatorio	Existencias principales	Recepción en blanco ...	Emisión en blanco pe...	Inventario físico	Inventario financiero	Plan de cobertura por dimensión
☑	Sitio	☑	☑	☐	☐	☑	☑	☑
☑	Almacén	☐	☐	☐	☐	☐	☐	☑
☑	Número de lote	☐	☐	☐	☐	☐	☐	☐
☑	Ubicación	☐	☐	☐	☐	☐	☐	☐
☐	Id de pallet	☐	☐	☐	☐	☐	☐	☐
☐	Número de serie	☐	☐	☐	☐	☐	☐	☐

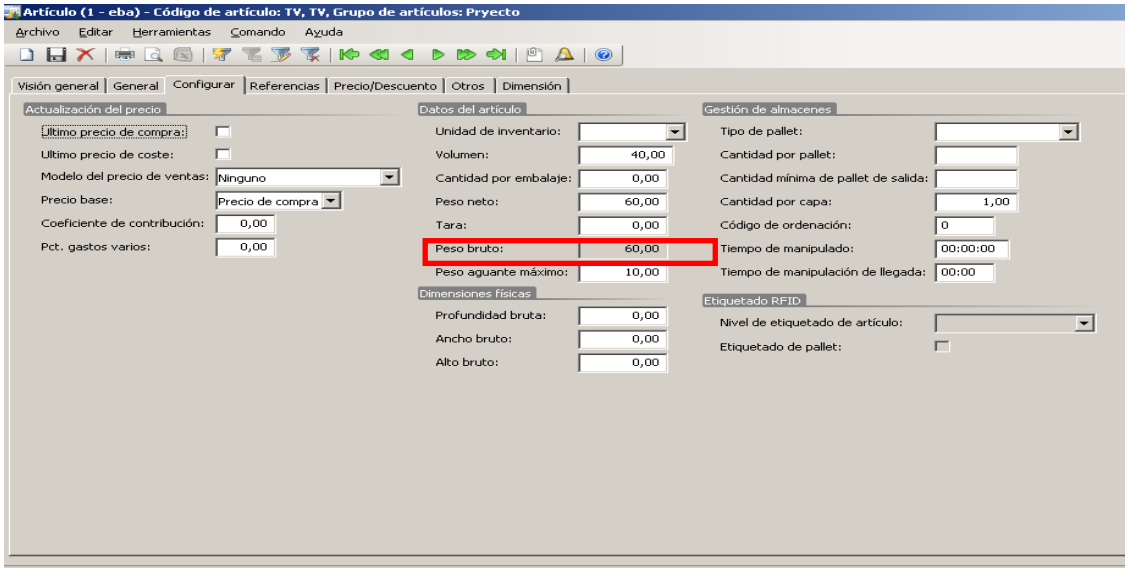
Il·lustració 32 Dimensions dels articles

Les dimensions per als articles del projecte són aquelles que tenen marcades el camp "Activo". De forma que, per a cada article caldrà informar del lloc on s'ha ubicat, del magatzem, del número de lot i de la ubicació concreta on està col·locat dins del magatzem.

5.4. Desenvolupaments nous en el mòdul d'inventari

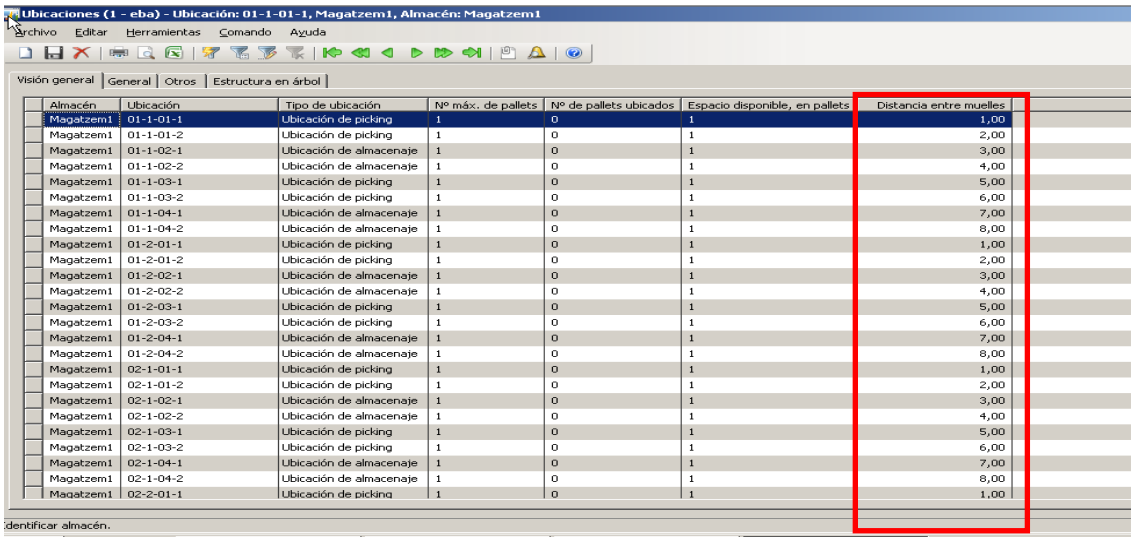
Com ja hem fet referència anteriorment, Axapta permet realitzar de forma fàcil qualsevol tipus de modificació o de procés nou. Per a dur a terme aquest algorisme s'han hagut de realitzar noves funcionalitats les quals s'esmenten en aquest capítol.

En el mestre d'articles s'ha introduït un camp nou anomenat "Peso aguante máximo". Aquest camp ens permet identificar per a cada article quin és el pes màxim que aquest és capaç de suportar damunt seu. Aquest valor s'ha de tenir en compte en el moment de recollir els articles i procedir a apilar-los.



Il·lustració 33 Pes d'aguant màxim de l'article

En el mestre d'ubicacions ha estat necessari crear un camp anomenat "Distancia entre muelles". Aquest camp ens permet introduir per a cada ubicació quina és la distància d'aquesta al moll de sortida. Aquest valor s'ha de tenir en compte en el moment de calcular el pes del camí i alhora d'escollir el millor camí per a la recollida dels productes.

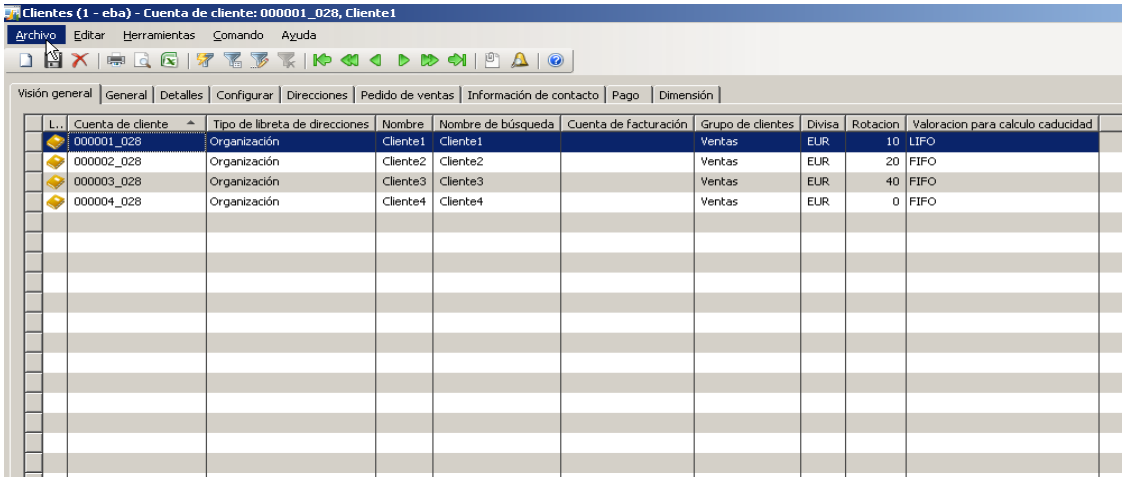


Almacén	Ubicación	Tipo de ubicación	Nº máx. de pallets	Nº de pallets ubicados	Espacio disponible, en pallets	Distancia entre muelles
Magatzem1	01-1-01-1	Ubicación de picking	1	0	1	1,00
Magatzem1	01-1-01-2	Ubicación de picking	1	0	1	2,00
Magatzem1	01-1-02-1	Ubicación de almacenaje	1	0	1	3,00
Magatzem1	01-1-02-2	Ubicación de almacenaje	1	0	1	4,00
Magatzem1	01-1-03-1	Ubicación de picking	1	0	1	5,00
Magatzem1	01-1-03-2	Ubicación de picking	1	0	1	6,00
Magatzem1	01-1-04-1	Ubicación de almacenaje	1	0	1	7,00
Magatzem1	01-1-04-2	Ubicación de almacenaje	1	0	1	8,00
Magatzem1	01-2-01-1	Ubicación de picking	1	0	1	1,00
Magatzem1	01-2-01-2	Ubicación de picking	1	0	1	2,00
Magatzem1	01-2-02-1	Ubicación de almacenaje	1	0	1	3,00
Magatzem1	01-2-02-2	Ubicación de almacenaje	1	0	1	4,00
Magatzem1	01-2-03-1	Ubicación de picking	1	0	1	5,00
Magatzem1	01-2-03-2	Ubicación de picking	1	0	1	6,00
Magatzem1	01-2-04-1	Ubicación de almacenaje	1	0	1	7,00
Magatzem1	01-2-04-2	Ubicación de almacenaje	1	0	1	8,00
Magatzem1	02-1-01-1	Ubicación de picking	1	0	1	1,00
Magatzem1	02-1-01-2	Ubicación de picking	1	0	1	2,00
Magatzem1	02-1-02-1	Ubicación de almacenaje	1	0	1	3,00
Magatzem1	02-1-02-2	Ubicación de almacenaje	1	0	1	4,00
Magatzem1	02-1-03-1	Ubicación de picking	1	0	1	5,00
Magatzem1	02-1-03-2	Ubicación de picking	1	0	1	6,00
Magatzem1	02-1-04-1	Ubicación de almacenaje	1	0	1	7,00
Magatzem1	02-1-04-2	Ubicación de almacenaje	1	0	1	8,00
Magatzem1	02-2-01-1	Ubicación de picking	1	0	1	1,00

Il·lustració 34 Distància de la ubicació al moll de sortida

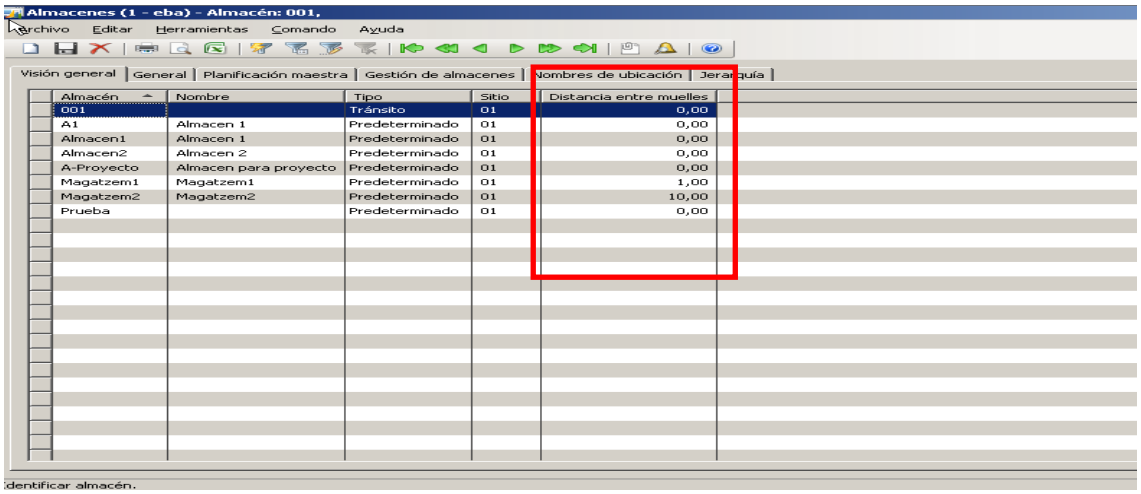
En el mestre de clients s'ha creat un camp anomenat "Rotación". Aquest camp és el que ens permet realitzar el càlcul del número de lot del client i, ens indica els dies que requereix el client per a què quan rebi el

producte aquest encara disposi d'un marge ampli de caducitat. Aquest marge és el que s'indica a la fitxa del client.



Il·lustració 35 Rotació del client

En el mestre de magatzems s'ha afegit el camp "Distancia entre muelles". Aquest camp ens permet especificar per a cadascun dels magatzems quina és la distància existent entre el moll d'entrada i el moll de sortida. Aquest valor també s'ha de tenir en compte per al càlcul de la millor ruta de recollida.

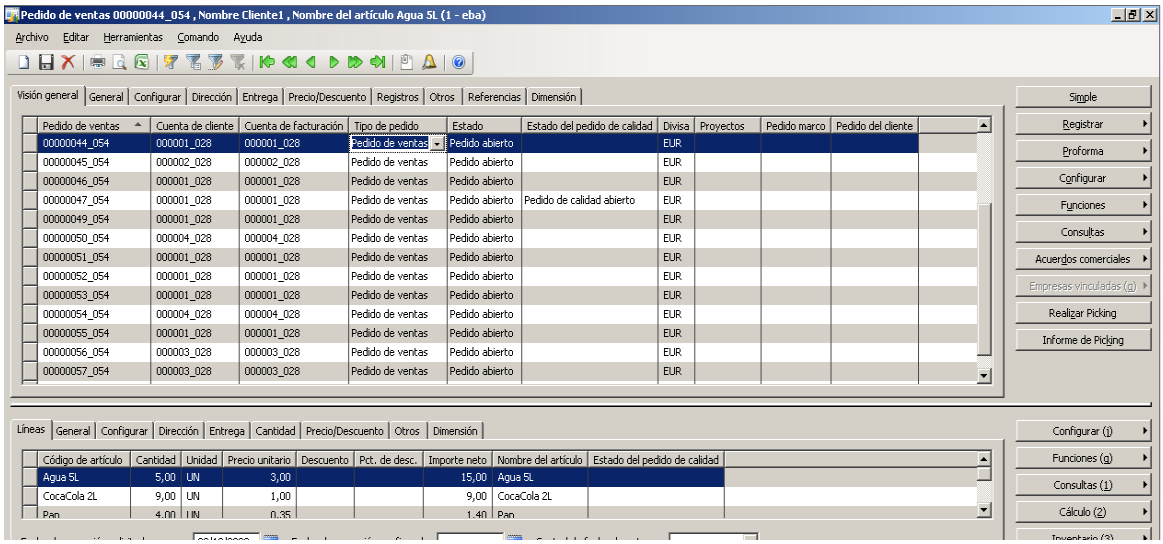


Il·lustració 36 Distància entre els molls del magatzem

A més a més, de la creació de camps en els diferents mestres esmentats, també ha estat necessari afegir la funcionalitat de crear el

Picking en el formulari de comandes de venta d' Axapta, per tal que, per a cada comanda de venta realitzada puguem llençar l'algorisme de Picking que s'ha implementat en aquest projecte.

Finalment, a part de la creació de l'algorisme que no es un procés estàndard de l'aplicació, s'ha realitzat un llistat que mostra el camí que ha de seguir l'usuari del magatzem per recollir els productes. En el formulari de comandes de venta, s'ha afegit una consulta a aquest llistat per veure per a cada comanda l'informe de Picking que s' entregarà a l'usuari.



Il·lustració 37 Funció per al Picking i imprimir llistat del Picking

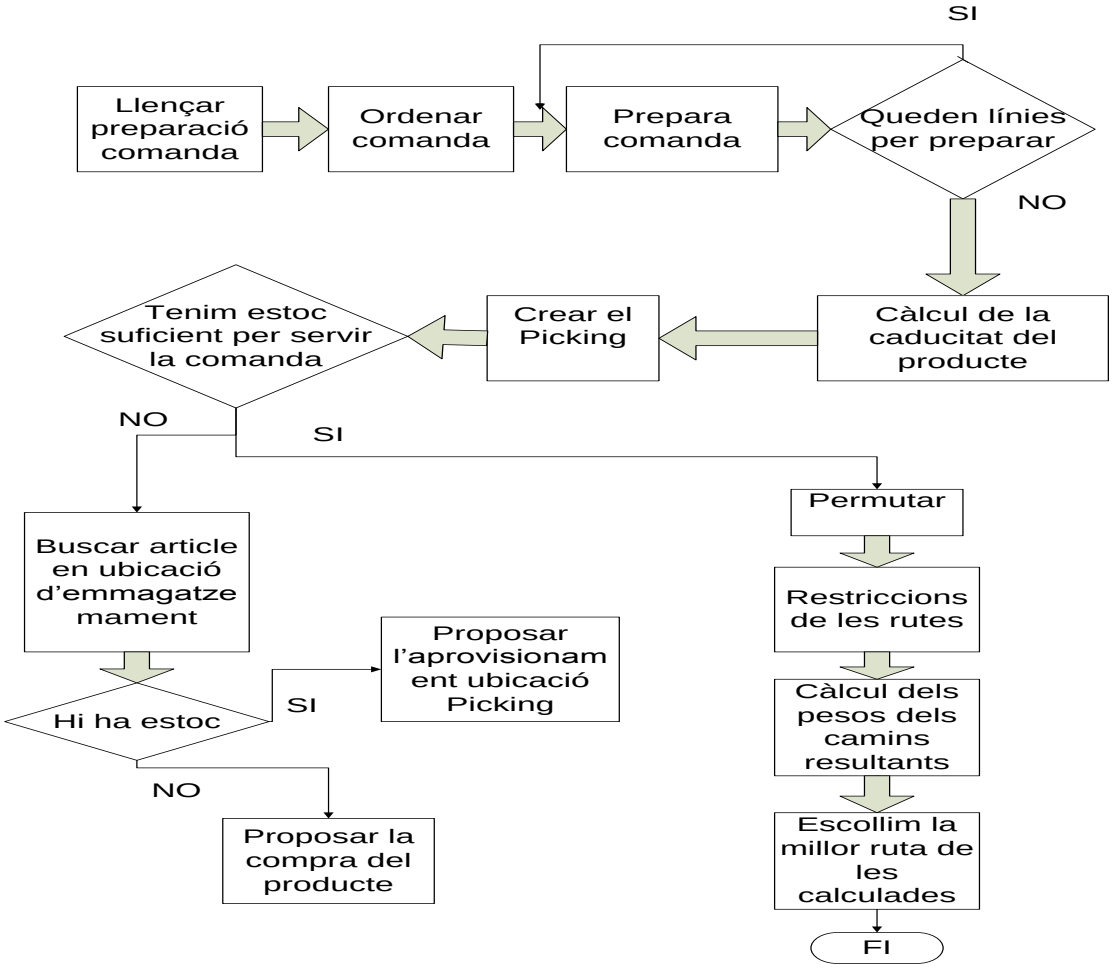
6. Funcionalitat de l'algorisme

En aquets capítol veurem de forma gràfica quin és el flux que segueix l'algorisme des de que s'inicia fins a que arriba al seu fi.

A més a més, explicarem cadascun dels mètodes / funcionalitats implementats en el procés.

6.1. Diagrama de flux

A continuació, veurem el diagrama de flux que segueix l'algorisme fins obtenir un resultat:



Il·lustració 38 Diagrama de flux

6.2. Funcionalitat dels mètodes de l'algorisme

En aquest apartat coneixerem els mètodes que s'han implementat per a l'execució de l'algorisme. A més a més, de cadascun d'aquests mètodes explicarem quina és la seva funcionalitat i en l'annex 1 en podrem veure el seu codi.

6.2.1. Main()

Aquest és el primer mètode de la classe. La seva finalitat consisteix en recollir per paràmetre el registre actiu[9] del formulari comandes de venta, per saber quins són els valors de la comanda que cal tenir en compte; així com, les línies que s'han de preparar. Seguidament es crida al següent mètode de la classe per realitzar la preparació de la comanda.

6.2.2. LanzarPreparaciónPedido()

Es selecciona de la taula InventTrans d'Axapta el número de comanda que s'ha recollit en el mètode Main(). Un cop accedim a la taula on tenim totes les dades de la comanda de venta i en la que s'ha de realitzar el Picking hem de guardar en una taula nova anomenada TmpArticulosPedido, els següents valors:

- Número de la comanda

- Articles a servir

- Quantitat a servir de l'article

- L'identificador de dimensió d'inventari per a controls i cerques internes a altres dades

[9] Registre actiu, element que hi ha seleccionat a la pantalla on es crida al procés de Picking. En aquest cas el registre actiu és la comanda de venta que s'ha de servir.

L'identificador d'inventari ens permet realitzar una cerca a la taula de Comandes de venta per obtenir:

- Data d'enviament sol·licitada

- Data de recepció sol·licitada pel client

També es realitza una cerca a la taula Dimensions mitjançant l'identificador de dimensió i, obtenim els camps que fan referència a la seva localització:

- Dimensió d'inventari, per a controls o cerques internes a altres dades

- Lloc de treball (un magatzem sempre té assignat un lloc de treball, de forma que N magatzems podrien estar al mateix lloc)

- Magatzem des del qual s'ha de servir la comanda

- Ubicació de Picking de l'article a servir

Busquem a la taula d'articles mitjançant el codi d'article per obtenir les característiques del producte:

- Pes

- Volum

- Pes d'aguant màxim

Finalment, es fa la crida al següent mètode per realitzar l'ordenació de la comanda, se li passa la taula que s'ha omplert en aquest mètode i el número d'operació declarat en el mètode Main().

6.2.3. OrdenarPedido()

Aquest mètode és l'encarregat d'ordenar la comanda de venta en funció del pes de l'article i del seu volum. L'ordenació es fa de forma descendent.

Ens guardem en una taula els següents camps:

- Ordre de recollida dels productes
- L'article a recollir
- La quantitat a servir
- El Magatzem
- La ubicació
- L'identificador de dimensió d'inventari
- Data d'enviament sol·licitada
- Data de recepció sol·licitada
- Codi de Client

- Lloc de treball

- Magatzem des d' on s'ha de servir la comanda

Aquests valors els guardem un altre cop i aquesta vegada ho fem en una taula que està ordenada en funció del pes i del volum de l'article. Aquesta taula la passarem al següent mètode (PreparaciónPedido()) per a què realitzi unes altres funcions.

6.2.4. PreparaciónPedido()

Aquest mètode, és l'encarregat de recórrer la taula que li ha passat el mètode anterior per realitzar la preparació de cadascuna de les línies de la comanda de venta.

Mentre hi hagin registres a la taula, aquest mètode calcula per a cadascuna d'elles, la data de caducitat per obtenir el número de lot de l'article que s'ha de servir. Per al càlcul de la caducitat del producte guardarem en una taula els següents valors:

- Camp valoració de la taula del client al qual li preparem la comanda

- Camp rotació de la taula del client al qual li preparem la comanda

- L'article

- La data d'enviament sol·licitada pel client

- La data de recepció sol·licitada pel client

El mètode que calcula la data de caducitat ens la retorna calculada per a cada producte en concret. En el mateix mètode i amb aquest valor

retornat es calcula quin és el número de lot; tenint en compte, el camp valoració especificat en el mestre de clients per al client en concret. En el cas, que el procés que calcula la data de caducitat no ens retorni res procedirem a preparar una altra línea de la comanda.

Si el mètode de valoració del client conté el valor FIFO, agafarem la data més propera als dies programats al càlcul de la caducitat i buscarem el número de lot que contingui aquesta data. En canvi, si el mètode de valoració conté el valor LIFO, agafarem la data més llunyana a la data calculada i buscarem també el número del lot que contingui aquesta data.

Finalment, guardarem a la taula CrearPicking les dades necessàries per dur a terme el següent procés que consisteix en crear el Picking. Els valors que es guarden són:

- L'article

- La quantitat a servir

- La ubicació de Picking de cada producte

- El lloc de treball

- El magatzem

La taula on guarden aquests valors es passa al següent mètode anomenat, CrearPicking.

6.2.5. CálculoCaducidad()

Aquest mètode ens retorna la data de caducitat que hauria de tenir l'article que s'ha d'enviar al client.

El primer càlcul consisteix en calcular els dies que tarda la mercaderia a arribar al client, tenint en compte, la data d'enviament sol·licitada i la data de recepció sol·licitada pel client que ja estan especificades en la comanda de venta.

Un cop saben els dies que tarden els productes des de que surten del magatzem fins que arriben al client, sumem el valor del camp rotació que té especificat el client en la seva fitxa per donar-li més marge de caducitat al producte, un cop ja el té el client.

Finalment, aquest mètode retorna la data de caducitat al mètode PreparacionPedido() per a què realitzi la recerca del número de lot que cal servir del producte tal i com s'ha explicat en el punt anterior.

6.2.6. CrearPicking()

Aquest mètode té com a finalitat consultar si disposem d'estoc disponible per servir la quantitat de l'article de la comanda de venda que s'està preparant per al Picking.

Sabent el número de lot del producte i la seva ubicació realitzem la consulta del disponible en el magatzem des del qual es serveix la comanda. Si tenim disponible guardem en la taula SegOperacions que s'ha de realitzar l'operació de Picking per al producte que s'està preparant en aquell moment i, especifiquem també el número de lot, la quantitat a servir, la ubicació de Picking i el magatzem des d' on servirem la comanda. En el cas de no haver-hi suficient quantitat del producte disponible en la ubicació de Picking, des d'aquest mètode, es crida una altra funció anomenada BuscarLoteUbicacionReposicion() que explicarem a continuació.

6.2.7. BuscarLoteUbicacionReposicion()

Aquest mètode es crida amb la finalitat de buscar en la ubicació d'emmagatzemament del producte si hi ha estoc disponible per servir la comanda.

La cerca es realitza en la taula d'ubicacions que conté Axapta per al magatzem corresponent i per al número de lot del producte.

En el cas de disposar de la quantitat suficient del producte, l'usuari del magatzem que està preparant la comanda haurà de realitzar la reposició a la ubicació de Picking i es fa la crida al mètode ReponerUbicacionPicking(). Si no hi ha la quantitat suficient del producte per servir la comanda guardem en la taula segOperacions que no es pot realitzar cap operació, ja que, no hi ha suficient quantitat per realitzar la reposició a la ubicació de Picking.

6.2.8. ReponerUbicacionPicking()

Guardem en la taula SegOperacions que cal fer reposició de la ubicació d'emmagatzemament a la ubicació de Picking del producte. A més a més, indiquem per a quin producte, per a quin número de lot, per a quin magatzem i per a quina ubicació de Picking.

Des d'aquest mètode, es fa la crida al mètode CrearPicking() per continuar realitzant la preparació de la comanda dels productes restants.

6.2.9 PesoUbicaciones()

Un cop sabem que podem realitzar l'enviament de tots els productes de la comanda del client, és a dir, que hi ha suficient disponible en la ubicació de Picking per als corresponents productes, comença la segona part d'aquest algorisme que consisteix en realitzar la cerca de la millor ruta de Picking.

En aquest mètode s'inicialitza en un vector totes aquelles ubicacions de Picking que l'usuari del magatzem a de recórrer per servir la comanda del client.

Un cop inicialitzat el vector, anomenat inicial, es realitza de forma consecutiva les crides als següents mètodes:

- Permutar()

- RestriccionRutas()

- CalcularPeso()

- EscogerMejorCamino()

6.2.9. Permutar()

És en aquesta funció on es realitzen les combinacions de totes les ubicacions que s'han inicialitzat en el mètode anterior. De forma que ens guardem en una taula, anomenada Permutar, totes les possibles combinacions de camins.

El següent pas que cal realitzar un cop tenim aquesta taula informada és la cerca del millor camí segons una sèrie de requisits. Aquesta cerca es realitza en el mètode RestriccionRutas().

6.2.10. RestricciónRutas()

Com ja hem comentat en el mètode anterior, la funcionalitat bàsica d'aquest consisteix, en recórrer tots els possibles camins de la taula on tenim totes les combinacions i escollir-ne un o varis depenent del pes que pot suportar cada producte damunt seu. Per fer-ho seleccionem un a un cada camí, ens posicionen en l'última ubicació del camí a recórrer per l'usuari (en l'última ubicació és on es col·locarà el producte que menys pes aguanti damunt seu) i buscarem per a l'article que hi ha en aquesta ubicació quin és el seu pes i el pes màxim que pot suportar damunt seu. En primer lloc, s'ha de realitzar una validació, a fi de, que no es pugui col·locar damunt del producte que estem recollint un altre producte que

tingui un pes major al que pot suportar. Guardem en una variable tipus real quin és el pes acumulat per al camí. En aquesta variable sumem el pes de l'article que estem recollint i el pes acumulat per al camí. En aquest cas, al ser el primer cop que es realitza el càlcul, el pes acumulat per a la ruta és 0 i, per tant entraria en la validació.

En el següent exemple, podrem veure de forma més clara en que consisteix aquest mètode:

- Característiques dels articles:

Article	Pes	Pes aguant del producte	Ubicació
A1	1	0,5	U1
A2	2	1,4	U2
A3	3	2	U3

- Possibles camins:

Camí 1	U1,U2,U3
Camí 2	U1,U3,U2
Camí 3	U2,U1,U3
Camí 4	U2,U3,U1
Camí 5	U3,U1,U2
Camí 6	U3,U2,U1

- Resultat del càlcul de l'algorisme:

  - Primer càlcul:

Ubicacions	Article	Pes producte	Pes aguant Producte	Pes Acumulat	Resultat
U1	A1	1	0,5	0	OK

U3	A3	3	2	1	OK
U2	A2	2	1,4	4	NO

Aquesta combinació de camí no seria valida, ja que, el pes acumulat per a la recollida de l'últim producte és superior al pes que pot suportar l'article A2.

- Segon càlcul:

Ubicacions	Article	Pes producte	Pes aguant Producte	Pes Acumulat	Resultat
U1	A1	1	0,5	0	OK
U2	A2	2	1,4	1	OK
U3	A3	3	2	3	OK

I així successivament, es realitzaria el càlcul per a la resta de combinacions de camins.

6.2.11. CalcularPeso()

Aquest mètode té per objectiu calcular el pes dels camins obtinguts en el mètode RestriccionRutas(). Fem la crida al primer mètode per calcular el pes dels camins anomenat, CalcularPesoPasillo().

6.2.12. ClcularPesoPasillo()

Realitzem una consulta en la taula "permutar" on hi han les combinacions dels camins. D'aquesta manera, per a cada ubicació s'obté les posicions de la ubicació que fan referència al passadissos, en aquest cas, seria les posicions 1 i 2.

El criteri per al càlcul del pes dels passadissos consisteix bàsicament, en realitzar la diferència entre cadascun dels passadissos de les ubicacions a recórrer, tenint en conte, un criteri lògic per a la distància que hi ha entre ells; és a dir, si per exemple: la primera ubicació es troba en el segon passadís i, la segona ubicació en el quart passadís la diferència en aquest

cas seria de 2. Cal recórrer doncs, cadascuna de les ubicacions per saber el pes total del camí per a la variable passadís. El total del pes del passadís ens el guardem en una variable anomenada "pesoPasillo" la qual s'utilitzarà per al càlcul del pes total. Aquesta variable del tipus enter és passa al següent mètode per calcular el pes dels prestatges i realitzar l'acumulat del pes al final del càlcul.

6.2.13. CalcularPesoEstanteria()

El criteri per al càlcul del pes dels prestatges es realitza de la mateixa manera que el que hem explicat en el mètode calcularPesoPassillo() però, en aquest cas, per obtenir les posicions del prestatge seleccionem de la ubicació la posició 4.

6.2.14. CalcularPesoBalda()

De la mateixa que en el cas anterior, el criteri per al càlcul del pes dels prestatges es realitza de la mateixa manera que el que hem explicat en el mètode calcularPesoPassillo() però, per obtenir les posicions del prestatge, en aquest cas, seleccionem de la ubicació les posicions 6 i 7.

En aquest mètode ja sabem quin és el pes total del camí. Ja que, tenim el pes dels passadissos, els dels prestatges i els de la balda. Es calcula el total del pes i es guarden els resultats en la taula "Pesos". A més a més, al total del pes també s'ha de tenir en compte el pes / distància que hi ha entre l'última ubicació a recórrer fins al moll de sortida que és on l'usuari acabarà la ruta de Picking. Cadascuna de les ubicacions té predeterminada una distància al moll de sortida que és la que es sumarà al pes total calculat fins al moment.

6.2.15. EscogerMejorCamino

Seleccionem de la taula "pesos" el camí que tingui el pes menor. En cas, d'haver-hi més d'un camí amb el mateix pes es selecciona el primer d'ells i, es guarda en la taula anomenada "TablaInforme" els valors

necessaris per al llistat de Picking que s'entregarà a l'usuari. Aquests valors són els següents:

- Ordre de la ruta (l'ordre en què l'usuari ha de recollir els productes)

- Codi d' article

- Quantitat

- Número de lot

- Ubicació

- Magatzem

7. Conclusió

Reflexionant sobre l'objectiu d'aquest estudi i basant-me en una sèrie de proves (les podem veure en l'annex número dos) que s'han realitzat per als dos dissenys de magatzems, podem concloure d'una manera molt clarificadora, que el millor disseny de magatzem és sens dubte el primer.

Les proves que s'han realitzat consisteixen en la creació de diferents comandes de venda per als respectius magatzems, de forma que, una mateixa comanda s'ha aplicat a tots dos magatzems per comprovar i analitzar quin són els resultats un cop aplicat l'algorisme d'optimització de rutes de Picking i de reposició.

Parlem doncs, que el millor disseny de magatzem és el primer ja que per a totes les proves realitzades hem pogut veure que els pesos dels camins en aquest cas han estat molt inferiors als pesos del camins del segon magatzem. És important remarcar que s'han realitzat les mateixes comandes de venda tant per un magatzem com per l'altre. Hem pogut observar com en el primer disseny de magatzem la distància que ha de recórrer l'usuari que prepara la comanda és molt inferior a la que hauria de recórrer en el segon disseny de magatzem estudiat en aquest projecte. Això és degut, a què les ubicacions del productes del primer magatzem són més properes al moll de sortida que en el segon cas. Per tant podem concloure, que la millor distribució per a un magatzem seria la primera que s'ha dissenyat.

Finalment, cal remarcar que els resultats esperats per a tots dos magatzems han estat els mateixos. És a dir, per a cada comanda de venda aplicada en cadascun dels magatzems l'ordre de recollida ha estat la mateixa, excepte en la tercera prova, on no ha estat així degut a que les característiques dels productes són molt similars i és indiferent escollir-ne primer un o l'altre.

8. Treballs futurs

Després de realitzar aquest projecte podem plantejar com a nova línea d'investigació la integració de la tecnologia de radio freqüència en el procés de Picking.

Aquesta nova via que es planteja consistiria en integrar la radio freqüència en el procés de Picking utilitzant un terminal mòbil, com per exemple: una PDA la qual utilitzaria l'usuari que està al magatzem preparant les comandes de venda que l'empresa serveix als seus clients. Aquest aparell mòbil s'alimentaria amb la informació que retorna el procés de Picking i de reposició d'Axapta i, de forma seqüencial informaria a l'usuari de totes les operacions que cal realitzar per a cadascuna de les comandes de venda que s'ha de servir durant el dia. Quan es parla de les operacions que ha de realitzar l'usuari estem fent referència per exemple a:

- Realitzar reposició d'un determinat producte.
- Fer el Picking d'un determinat producte.

9. Bibliografia

Manual de consulta de Microsoft Dynamics AX 2009:

- [Axapta *Introduction] Micrososft Dynamics Corporation, Microsoft ® Business Solutions – Axapta ® Introduction*

Manual de consulta de gestió de magatzems de Microsoft Dynamics AX 2009:

- [Axapta Gestió Inventari*] Micrososft Dynamics Corporation, Microsoft ® Business Solutions – Axapta ® Wharehouse Management in Microsoft Dynamics AX 2009*

Manual de consulta de MorphX de Microsoft Dynamics AX 2009:

- [Axapta *MorphX] Micrososft Dynamics Corporation, Microsoft ® Business Solutions – Axapta* ® Guia de Referència MorphX

Manual de consulta de MorphX de Microsoft Dynamics AX 2009:

- [Axapta *X++] Micrososft Dynamics Corporation, Microsoft ® Business Solutions – Axapta* ® Guia de Referència X++

Referències Web:

- [Entorn Morphx]Entorn Morph:
    - http://www.ideonetworks.com/index.php?/dynamics-ax/morphx-y-x.html
- [DragDrop] Significat de Drag and Drop:
    - http://www.es.wikipedia.org/wiki/Arrastrar_y_soltar
- [Assentament] Significat Assentament contable:
    - http://www.cecot.es/documents/Doc676.pdf
- [ESTC] Estructura genèrica d'un magatzem:
    - http://www.marianoseral.com/almacen/estructura.asp
- [TIPO] Tipologies d'inventari:
    - http://www.monografias.com/trabajos15/inventario/inventario.shtml
- [PICK] Definició del Picking:
    - http://es.wikipedia.org/wiki/Picking

Treball Final de Carrera

Implementació d'un algorisme d'optimització de rutes de Picking i reposició de magatzems

ANEXOS

Annex 1: Visualització dels mètodes de l'algorisme

Main()

```
static void main(args _args)
{
    int                     numOp;
    Pesos                   pesos;
    SegOperacions           segOperacions;
    EstocDisponible         estocdisponible;
    Permutar                permutar;
    TablaInforme            tablaInforme;
    CrearPicking            crearPicking;
    TmpPedidosOrdenados     tmpPedidosOrdenados;
    Reposicion              reposicion;
    TmpArticulosPedido      tmpArticulosPedido;
    TmpCaducidad            tmpCaducidad;
    SalesTable              st;
    SalesIdBase             numeroPedido;
    ;

    delete_from     pesos;
    delete_from     TablaInforme;
    delete_from     segOperacions;
    delete_from     estocdisponible;
    delete_from     permutar;
    delete_from     crearPicking;
    delete_from     tmpPedidosOrdenados;
    delete_from     reposicion;
    delete_from     tmpArticulosPedido;
    delete_from     tmpCaducidad;


    //Recolleix el registre actiu de la taula de Pedidos de Venta

        select st where _args.record().RecId == st.RecId;
            numeroPedido = st.SalesId;

        numOp = 1;
        pickingFinal::LanzarPreparacion(numOp, numeroPedido);
        PickingFinal::PesoUbicaciones();
}
```

LanzarPreparaciónPedido()

```
static void LanzarPreparacion(int numOp, SalesIdBase numeroPedido)
{
    SalesLine                           salesLine;
    SalesTable                          salesTable;
    TmpArticulosPedido                  tmpArticulosPedido;
    InventDim                           inventDim;
    InventTable                         inventTable;
    InventBatch                         inventBatch;
    InventTrans                         inventTrans;
    SalesId                             salesIdOld;
    WmsLocationType                     wmsLocationType;
    boolean                             preparaPedido = false;
    ;

     //Seleccionem les comandes que estan llestes per a realitzar el picking

    while select * from InventTrans
        order by TransRefId
            where inventTrans.TransType == InventTransType::Sales
                        && inventTrans.TransRefId == numeroPedido

    {
        TmpArticulosPedido.NumPedido                        = inventTrans.TransRefId;
        TmpArticulosPedido.ItemId                           = inventTrans.ItemId;
        TmpArticulosPedido.Qty                              = inventTrans.Qty;
        TmpArticulosPedido.InventDimId                      = inventTrans.inventDimId;

        select SalesTable where salesTable.SalesId == inventTrans.TransRefId;
        TmpArticulosPedido.AccountNum = SalesTable.CustAccount;
        select * from SalesLine where SalesLine.SalesId == TmpArticulosPedido.NumPedido;

            //Guardem la data d'enviament sol·licitada
            TmpArticulosPedido.SalesReceiptDateRequested        = salesLine.ReceiptDateRequested;
            //Guardem la data de recepcio del cliente sol·licitada
            TmpArticulosPedido.SalesShippingDateRequested       = salesLine.ShippingDateRequested;
            //Guardem la dimensió d'inventari, per a consultar el magatzem
            inventDim                                           = inventDim::find(inventTrans.inventDimId);
            //Guardem el lloc de treball
            TmpArticulosPedido.InventSiteId                     = inventDim.InventSiteId;
            //Guardem el magatzem
            TmpArticulosPedido.InventLocationId                 = inventDim.InventLocationId;
            //Guardem l'ubicación de Picking
            TmpArticulosPedido.WMSLocationId                    = inventDim.wMSLocationId;

            //Seleccionem per al article corresponent el pes, volum i els guardem a la taula temporal
            inventTable                                         = inventtable::find(inventTrans.ItemId);
            TmpArticulosPedido.ItemNetWeight                    = inventTable.NetWeight;
            TmpArticulosPedido.Volumen                          = inventTable.UnitVolume;
            TmpArticulosPedido.PesoEncima                       = inventTable.PesoEncima;

            preparapedido = true;
            TmpArticulosPedido.insert();

    }

    PickingFinal::OrdenarPedido(TmpArticulosPedido,numOp);
}
```

OrdenarPedido()

```
static void OrdenarPedido(TmpArticulosPedido TmpArticulosPedido, int numOp)
{
    TmpPedidosOrdenados          tmpPedidosOrdenados;
    InventTable                  inventTable;
    str                          medida;
    weight                       weight;
    int                          i = 0;
    InventDimId                  inventDimOld;
    Int                          pesoMaximoEncima;
    ;

     // Primer ordenem la taula segons el pes màxim d'aguant de l'article
    //Hem de recollir primer aquells articles que tinguin més aguant
   // Despres ordenarem segons el volum de cada article

    ttsbegin;
    while select TmpArticulosPedido
        order by TmpArticulosPedido.PesoEncima desc,
            TmpArticulosPedido.Volumen desc

    {

            //Si un article pot aguantar 3kg damunt seu no podem posar-li un artículo que pesi 60kg damunt seu

            TmpPedidosOrdenados.Ordenacion                = i+ 1;
            TmpPedidosOrdenados.ItemId                    = TmpArticulosPedido.ItemId;
            TmpPedidosOrdenados.Qty                       = TmpArticulosPedido.Qty;
            TmpPedidosOrdenados.WMSLocationId             = TmpArticulosPedido.WMSLocationId;
            TmpPedidosOrdenados.InventDimId1              = TmpArticulosPedido.InventDimId;
            inventDimOld                                  = TmpArticulosPedido.InventDimId;

            tmpPedidosOrdenados.SalesReceiptDateRequested   = TmpArticulosPedido.SalesReceiptDateRequested;
            tmpPedidosOrdenados.SalesShippingDateRequested  = TmpArticulosPedido.SalesShippingDateRequested;
            tmpPedidosOrdenados.AccountNum                  = TmpArticulosPedido.AccountNum;
            tmpPedidosOrdenados.InventSiteId                = TmpArticulosPedido.InventSiteId;
            tmpPedidosOrdenados.InventLocationId            = TmpArticulosPedido.InventLocationId;

            TmpPedidosOrdenados.insert();

            i = tmpPedidosOrdenados.Ordenacion;

    }
    ttscommit;
    //Fins aquí tenim la taula TmpPedidosOrdenada en l'ordre de recollida preferent, faltarà veure quin es el millor camí el pes d'aguant i el pes de l'article.

    PickingFinal::PreparacionPedido(tmpPedidosOrdenados);
}
```

PreparacionPedido()

```
static void PreparacionPedido(TmpPedidosOrdenados tmpPedidosOrdenados)
{
    custTable           custTable;
    tmpCaducidad        tmpCaducidad;
    InventBatch         inventBatch;
    tmpPicking          tmpPicking;
    InventDim           inventDim;
    wmsLocation         wmsLocation;
    InventDimId         inventDimCaducidad;
    CrearPicking        crearPicking;

    ;

    ttsbegin;
    //Tenim que controlar si encara existeixen líneas en la taula temporal (tmpPedidosOrdenados) per a la seva preparació
    while select tmpPedidosOrdenados
    {
        //Comprobamos si aun existen líneas para preparar
        if ( !tmpPedidosOrdenados.ItemId)
        {

            //Si no existen líneas de pedido, ya hemos finalizado la preparación

            info("El proceso de preparación de pedidos ha terminado");

            PickingFinal::PesoUbicaciones();
        }
        else
        {
            //Si encara existeixen línies per preparar, les tenim que preparar per a procedir amb el Picking.
            //Primerament tenim que saber quin es el número de lot de l'article a servir.
            //Per a saber el número de Lot tenim que realitzar el càlcul de la caducitad de l'article, tenint en conte
            //un paràmetre dels clients, anomenat Rotació.També tenim que tenir en compte la dimensió d'inventari
            //FIFO o LIFO, per al càlcul.


            //Per a la realització del càlcul tenim que consultar la valoració i rotació del client de la comanda

            custTable                               = custTable::find(tmpPedidosOrdenados.accountNum);

            //Guardem la valoració segons client
            tmpCaducidad.Valoracion                 = custTable.Valoracion;
            //Guardem la rotació segons client
            tmpCaducidad.Rotacion                   = custTable.Rotacion;
            //Guardem l'article
            tmpCaducidad.ItemId                     = tmpPedidosOrdenados.ItemId;
            //Guardem el número lot de l'article
            tmpCaducidad.InventBatchId              = tmpPedidosOrdenados.InventBatchId;
            //Guardem la data d'enviament sol·licitada
            tmpCaducidad.SalesShippingDateRequested = tmpPedidosOrdenados.SalesShippingDateRequested;
            //Guardem la data de recepció sol·licitada
            tmpCaducidad.SalesReceiptDateRequested  = tmpPedidosOrdenados.SalesReceiptDateRequested;

            tmpCaducidad.insert();

            tmpCaducidad.expDate       = PickingFinal::CalculoCaducidad(TmpCaducidad);
```

```
if (tmpCaducidad.expDate)
{
    //Tenim que guardar el número de lot adequat segosn el càlcul de la data de caducitat
    // a la taula tmpArticulosMismoPedido

    if (tmpCaducidad.Valoracion == valoracion::FIFO)
    {
        // Si es FIFO, agafem la darta mes propera als dies programats
        select firstonly inventbatchid
            from InventBatch
                order by expdate asc
                    where tmpCaducidad.expDate <= inventBatch.expDate
                        && inventBatch.itemId == tmpPedidosOrdenados.ItemId;

            //Seleccionem de l'article concret, el número de lot que contingui una data major a la data
            //calculada per a la caducitat
            CrearPicking.InventBatchId            = inventBatch.inventBatchId;
            TmpCaducidad.InventBatchId            = inventBatch.inventBatchId;


            select firstOnly inventDim
                index hint BatchIdIdx
                where inventDim.inventBatchId          == CrearPicking.InventBatchId
                   && inventDim.InventLocationId       == tmpPedidosOrdenados.InventLocationId
                join wmsLocation
                    index hint LocationIdx
                    where wmsLocation.wMSLocationId    == inventDim.wMSLocationId
                       && wmsLocation.inventLocationId == inventDim.InventLocationId
                       && wmsLocation.locationType     == wmsLocationType::Pick;

            InventDimCaducidad            = inventDim.inventDimId;

    }
    else
    {
        // Si es LEFO, agafem la data més llunyana als dies calculats
        select firstonly inventbatchid
            from InventBatch
                order by expdate desc
                    where tmpCaducidad.expDate <= inventBatch.expDate
                        && inventBatch.itemId == tmpPedidosOrdenados.ItemId;

            //Seleccionem de l'article concret, el número de lot que contingui una data major a la data
            //calculada para a la caducitat
            CrearPicking.InventBatchId      = inventBatch.inventBatchId;

            select firstOnly inventDim
                index hint BatchIdIdx
                where inventDim.inventBatchId          == CrearPicking.InventBatchId
                   && inventDim.InventLocationId       == tmpPedidosOrdenados.InventLocationId
                join wmsLocation
                    index hint LocationIdx
                    where wmsLocation.wMSLocationId    == inventDim.wMSLocationId
                       && wmsLocation.inventLocationId == inventDim.InventLocationId
```

```
                InventDimCaducidad              = inventDim.inventDimId;
                CrearPicking.InventDimId        = InventDimCaducidad;


            }

            //Guardem la informació necessaria en la tyaula de CrearPciking per a continuar amb el procés de Pciking.
            CrearPicking.ItemId             = tmpPedidosOrdenados.ItemId;
            CrearPicking.Qty                = tmpPedidosOrdenados.Qty;
            CrearPicking.WMSLocationId      = tmpPedidosOrdenados.WMSLocationId;
            CrearPicking.InventSiteId       = tmpPedidosOrdenados.InventSiteId;
            CrearPicking.InventLocationId   = tmpPedidosOrdenados.InventLocationId;

            CrearPicking.insert();

            PickingFinal::CrearPicking(crearPicking.ItemId, crearPicking.InventDimId, crearPicking.Qty,crearPicking);
        }
    }
  }
  ttscommit;
}
```

CalculoCaducidad()

```
static transdate CalculoCaducidad(tmpCaducidad tmpCaducidad)
{
    //Consultar el camp rotació client i el mètode de valoració

    int                 factor;
    int                 DiasVencimiento;
    Dimension           Dimension;
    int                 DiasEnLlegar; // Dias que tarda la mercancia en llegar al cliente.
    int                 DiasMargen;
    Date                EjFechaCaducidad;

    ;



        //Calculem els dies que tarda la mercaderia en arribar al client
        DiasEnLlegar   = tmpCaducidad.SalesShippingDateRequested - tmpCaducidad.SalesReceiptDateRequested;

        // Calculem el marge per a la data de caducitat des de que el client rep el producte
        factor         = tmpCaducidad.Rotacion;

        DiasMargen     = DiasEnLlegar + factor;

        EjFechaCaducidad = tmpCaducidad.SalesReceiptDateRequested + DiasMargen;

       return EjFechaCaducidad;

}
```

CrearPicking()

```
static void CrearPicking(ItemId _itemId, InventDimId _inventDimId, Qty _qty, crearPicking   crearPicking)
{
    InventSumDatePhysicalDim            inventSumDatePhysicalDim;
    Qty                                 disponible;
    transDate                           hoy;
    InventDim                           inventDim;
    InventDimId                         inventDimId;
    InventDimParm                       inventDimParm;
    EstocDisponible                     EstocDisponible;
    InventBatchId                       NumLoteServir;
    Qty                                 cantidadServirRestante;
    Qty                                 cantidadServir;
    InventlocationId                    almacen;
    Operacion                           Operacion;
    SegOperacions                       segOperacions;
    ;

    hoy = today();
    cantidadServir      = _qty;
    inventDim           = inventDim::find(_inventDimId);
    NumLoteServir       = inventDim.inventBatchId;

    inventDimParm.initFromInventDim(inventDim);


    //Busquem el disponible de l'article en funció del número de lot i de l'ubicació
    disponible       = inventSumDatePhysicalDim::onHandQty(hoy,_itemid,inventDim,inventDimParm);

    almacen = crearPicking.InventLocationId;

    //Tenim que mirar si podem servir tota la mercaderia de l'ubicació de Picking, si no tenim que anar a buscar l'article a la ubicació de reposició

    if (-(crearPicking.Qty) > disponible)
    {
        //No pdem servir del mateix lot, ja que no hi ha suficientment estoc en ubicació de Picking
        //Tenim que buscar en ubicació de reposició si enim la quantiat que nos falta per servir amb el lot concret

        cantidadServirRestante = -(cantidadServir) - Disponible;

        PickingFinal::BuscarLoteUbicacionReposicion(NumLoteServir,cantidadServirRestante,_ItemId,_InventDimId,cantidadServir,almacen);
    }

        //Podem servir al client. Tenim quantiat disponible suficient.

        //Restem al disponible de l'article la quanitat que preparem per al client
        //Tenim que guardar el disponible en alguna de les taules i consultar per al mateix pedido el disponible en aquesta taula.

        disponible = disponible - (-cantidadServir);

        EstocDisponible.ItemId              = crearPicking.ItemId;
        EstocDisponible.InventBatchId       = crearPicking.InventBatchId;
        EstocDisponible.Qty                 = disponible;

        //Omplim la taula doperacions, per saber en tot moment que es el que tenim que fer, es a dir, quina operació.

        select maxof(OrdenOp) from SegOperacions;

            if(SegOperacions.ordenOp == 0)
            {
                SegOperacions.OrdenOp = 1;
            }
            else
            {
                SegOperacions.OrdenOp = SegOperacions.OrdenOp + 1;
            }

        SegOperacions.Operacion         = operacion::Picking;
        SegOperacions.ItemId            = crearPicking.ItemId;
        SegOperacions.InventBatchId     = crearPicking.InventBatchId;
        SegOperacions.Qty               = CantidadServir;
        SegOperacions.WMSLocationId     = crearPicking.WMSLocationId;
        SegOperacions.InventLocationId  = almacen;
        SegOperacions.InventDimId       = _inventDimId;


        EstocDisponible.insert();
        SegOperacions.insert();


        //Seguim amb la preapració de la comanda

}
```

BuscarLoteEnUbicacionReposicion()

```
static void BuscarLoteUbicacionReposicion(InventBatchId _NumLoteServir, qty _CantidadServirRestante,ItemId _ItemId,InventDimId _InventDimId,Qty cantidadServir,
                                          InventLocationId almacen)
{
    WmsLocation             wmsLocation;
    InventBatchId           numLoteServir;
    Qty                     cantidadServirRestante;
    Qty                     disponible;
    InventDim               inventDim;
    InventDimParm           inventDimParm;
    WMSLocationType         wmsLocationType;
    TransDate               hoy;
    WMSLocationId           ubicacionAlmacenaje;
    boolean                 parar;
    Operacion               operacion;
    SegOperacions           SegOperacions;
    ;

    cantidadServirRestante      = _cantidadServirRestante;

    hoy             = today();

    //Tenim que buscar en l'ubicació d'emmagazemament si hi ha estoc suficient per a servir la petició del client amb el mateix número de lot
    //Buscar per a totes les ubicaciones d'emmagazemament si existeix disponible amb el Lot específic per al magatzem concret de la comanda

    while select wmslocationId from inventDim where inventDim.inventBatchId == _NumLoteServir
     && inventDim.inventlocationId == almacen
    {
        while select * from wmsLocation where wmslocation.inventLocationId == almacen
            && inventdim.wMSLocationId == wmslocation.wMSLocationId
            && wmsLocation.LocationType == wmsLocationType::Buffer

        {
            if ( wmsLocation.LocationType == wmsLocationType::Buffer)
            {
                ubicacionAlmacenaje = wmslocation.wMSLocationId;
                parar = true;
            }

        }
        if(parar == true)
            break;
    }

    //En aquest punt tenim l'ubicació d'emmagatzemament i tenim que buscar si tenim dispobible
    inventDim           = inventDim::find(_inventDimId);
    NumLoteServir       = inventDim.inventBatchId;

    inventDimParm.initFromInventDim(inventDim);

    disponible      = inventSumDatePhysicalDim::onHandQty(hoy,_itemid,inventDim,inventDimParm);

    if (cantidadServirRestante < disponible)
    {
        //Tenim que crear reposició per a l'ubicació de Picking i finalment servir el producte
        PickingFinal::ReponerUbicacionPicking(_NumLoteServir,_CantidadServirRestante,_ItemId,_InventDimId,cantidadServir, ubicacionalmacenaje, almacen, _inventDimId);

    }

    else
    {

            select maxof(OrdenOp) from SegOperacions;

             if(SegOperacions.ordenOp == 0)
             {
                 SegOperacions.OrdenOp = 1;
             }
             else
             {
                 SegOperacions.OrdenOp = SegOperacions.OrdenOp + 1;
             }

        //Insertar a la taula d'opreacions que no es pot realitzar cap operació
        // ja que no existeix disponible per a picking ni en l'ubicació de reposició

        SegOperacions.Operacion         = operacion::NoOp;
        SegOperacions.ItemId            = _ItemId;
        SegOperacions.InventBatchId     = _NumLoteServir;
        SegOperacions.Qty               = _CantidadServirRestante;
        SegOperacions.WMSLocationId     = ubicacionalmacenaje;
        SegOperacions.InventLocationId  = almacen;
        SegOperacions.InventDimId       = _inventDimId;

        SegOperacions.insert();

    }
    //si no podem servir el número de lot concret de l'article tenim que seguir amb la preparació de la comanda

}
```

ReponerUbicacionPicking()

```
static void ReponerUbicacionPicking(InventBatchId _NumLoteServir, qty _CantidadServirRestante,ItemId _ItemId,
                                    InventDimId _InventDimId,Qty _cantidadServir,
                                      wmslocationId ubicacionalmacenaje,InventLocationId almacen, InventDimId inventDimId)
{
    Estocdisponible             EstocDisponible;
    Operacion                   operacion;
    SegOperacions               SegOperacions;
    ;


    //Indiquem al sistema que cal fer reposició




     select maxof(OrdenOp) from SegOperacions;

            if(SegOperacions.ordenOp == 0)
            {
                SegOperacions.OrdenOp = 1;
            }
            else
            {
                SegOperacions.OrdenOp = SegOperacions.OrdenOp + 1;
            }

     SegOperacions.Operacion        = operacion::Reponer;
     SegOperacions.ItemId           = _ItemId;
     SegOperacions.InventBatchId    = _NumLoteServir;
     SegOperacions.Qty              = _CantidadServirRestante;
     SegOperacions.WMSLocationId    = ubicacionalmacenaje;
     SegOperacions.InventLocationId = almacen;
     SegOperacions.InventDimId      = _inventDimId;
     SegOperacions.OrdenOp           = segOperacions.OrdenOp +1;


     SegOperacions.insert();


    PickingFinal::CrearPicking(_itemId,_inventDimId,_CantidadServir, null);
}
```

PesoUbicaciones()

```
static void CalcularPeso()
{
    int totalPeso;
    int PesoPasillo;
    int PesoEstanteria;
    int PesoBalda;
    int PesoHueco;

    int i;
    Permutar    permutar;
    ;

    //Tenemos que seleccionar cada uno de los registros de la tabla para calcular peso en cada fila y sus combinaciones
    pickingFinal::CalcularPesoPasillo();

}
```

Permutar()

```
static void Permutar(container _c, integer _pos)
{
    int i,n;
    container   aux = _c;
    AnyType     temp;
    Anytype     v;
    Permutar    permutar;
    ;

    if (_pos == 1)
    {
        //Escriure la primera combincació a la taula Permutar
        permutar.Combinacion = aux;
        permutar.insert();
    }
    else
    {
        //Permutem els camins posibles en funció de les ubicacions que tenim que recorrer
        while (i < _pos)
        {
            i++;
            Prueba::Permutar(aux, _pos - 1);
            temp = conPeek(aux, i);
            aux  = conPoke(aux, i, conPeek(aux, _pos));
            aux  = ConPoke(aux, _pos, temp);
        }
    }

}
```

RestriccionRutas()

```
static void RestriccionRutas(Permutar _permutar)
{
    TmpArticulosPedido      tmpArticulosPedido;
    Permutar                permutar;
    container               c;
    WmsLocationId           cadena;
    real                    PesoAcum;
    real                    pesoArticulo, pesoEncima;
    int                     n,i;
    ;

    //Tenim que recorre tots els possibles camins de la taula Permutar per a veure quin escollim depenent del pes que poden soportar cada article

    select count(RecId) from tmpArticulosPedido;

    i = tmpArticulosPedido.RecId;

    ttsbegin;
    while select forupdate permutar
       order by RecId

    {
        n = i;
        c = permutar.Combinacion;

        while ( n != 0)
        {
            cadena = conpeek(c,n);

            select * from tmpArticulosPedido where tmpArticulosPedido.wMSLocationId == cadena;

            pesoEncima      = tmpArticulosPedido.PesoEncima;
            pesoArticulo    = tmpArticulosPedido.ItemNetWeight;

             //si el pes acumulat es major que el pes que pot soportar l'article es descartaà el camí
            if (pesoAcum >= pesoEncima)
            {
               permutar.delete(); //Eliminem la ruta
               n = 0;
               pesoAcum = 0;

            }
            else
            {
                //Si el pes acumulat no es major continuem recorrent la ruta
                //Acumulem el pes de l'article que recollim en el pes acumulat de la ruta
                pesoAcum = pesoAcum + PesoArticulo;

                if (n == 1)
                    PesoAcum = 0;
                n--;
            }
        }

    }
    ttscommit;
}
```

CalcularPeso()

```
static void CalcularPeso()
{
    int totalPeso;
    int PesoPasillo;
    int PesoEstanteria;
    int PesoBalda;
    int PesoHueco;

    int i;
    Permutar    permutar;
    ;

    //Tenemos que seleccionar cada uno de los registros de la tabla para calcular peso en cada fila y sus combinaciones
    pickingFinal::CalcularPesoPasillo();

}
```

CalcularPesoPasillo()

```
static void CalcularPesoPasillo()
{
    int             PesoPasillo,PesoEstanteria;
    Permutar        permutar;
    Pesos           pesos;
    int i = 1,j,z=1, diferencia;
    int num;
    container c,aux;
    str cadena1,cadena2;
    int x = 2;
    ;

    while select permutar
    order by recId
    {
        c = permutar.Combinacion;

        for (i=1; i < conlen(c); i++)
        {
            cadena1 = conpeek(c,i);
            cadena2 = conpeek(c,i+1);

            num = str2int(substr(cadena1,0,2));
            num = num - str2int(substr(cadena2,0,2));

            num = abs(num);

            PesoPasillo = PesoPasillo + num;

            if ( i == conLen(c) - 1)
            {

                //Busquem pes estanteria

                PickingFinal::CalcularPesoEstanteria(c, pesoPasillo);
                PesoPasillo = 0;
            }
        }
    }
}
```

CalcularPesoEstanteria()

```
static void  CalcularPesoEstanteria(container c, int _pesoPasillo)
{
    int             PesoEstanteria, pesoPasillo;
    Pesos           pesos;
    int i = 1,j,z=1, diferencia;
    int num;
    container aux;
    str cadena1,cadena2;

    ;
    pesoPasillo = _pesoPasillo;
    for (i=1; i < conlen(c); i++)
    {
        cadena1 = conpeek(c,i);
        cadena2 = conpeek(c,i+1);

        num = str2int(substr(cadena1,4,1));
        num = num - str2int(substr(cadena2,4,1));

        num = abs(num);

        PesoEstanteria = PesoEstanteria + num;

        if ( i == conLen(c) - 1)
        {
           //Busquem pes balda
            Pesos.PesoEstanteria = PesoEstanteria;
            PickingFinal::CalcularPesoBalda(c, pesoPasillo, pesoEstanteria);
            PesoEstanteria = 0;
        }
    }
}
```

CalcularPesoBalda()

```
static void  CalcularPesoBalda(container c, int _pesoPasillo, int _PesoEstanteria)
{
    int         PesoPasillo, PesoEstanteria, PesoBalda, PesoTotal1, pesototal2;
    Pesos       pesos;
    int i = 1,j,z=1, diferencia;
    int num;
    container aux;
    PickingFinal    PickingFinal;
    str cadena1,cadena2;

    ;
    pesoPasillo         = _pesoPasillo;
    pesoEstanteria      = _pesoEstanteria;

    PickingFinal = new pickingFinal();


    for (i=1; i < conlen(c); i++)
    {
        cadena1 = conpeek(c,i);
        cadena2 = conpeek(c,i+1);

        num = str2int(substr(cadena1,6,2));
        num = num - str2int(substr(cadena2,6,2));

        num = abs(num);

        PesoBalda = PesoBalda + num;

        if ( i == conLen(c) - 1)
        {

            Pesos.PesoBalda = PesoBalda;

            Pesos.PesoEstanteria = PesoEstanteria;

            Pesos.PesoPasillo = PesoPasillo;

            PesoTotal1 = PesoBalda + PesoEstanteria + PesoPasillo;

            PesoTotal2 = Pickingfinal.CalcularPesoMuelle(pesoTotal1,c);

            pesos.TotalPeso = pesoTotal2;

            Pesos.combinacion = c;

            Pesos.insert();

            PesoBalda = 0;
        }
    }

}
```

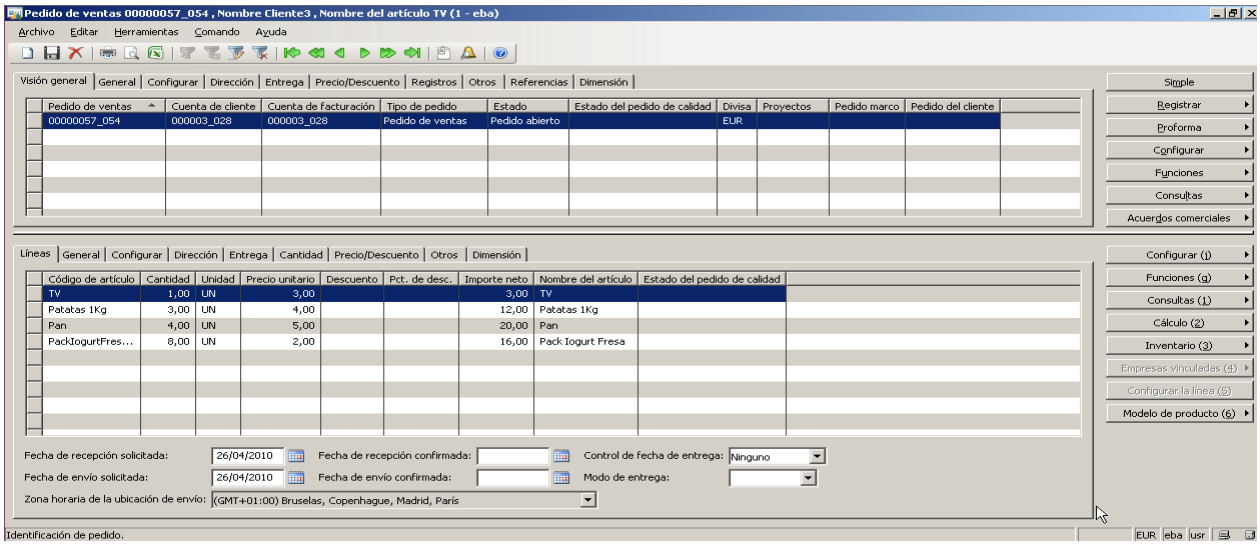
EscogerMejorCamino()

```
static void EscogerMejorCamino()
{
    Pesos               pesos;
    int                 pesoMinimo;
    int                 numOp = 0;
    int                 i = 1;
    container           aux;
    anytype             v;
    str                 cadena;
    TmpArticulosPedido  tmpArticulosPedido;
    TablaInforme        tablaInforme;
    ItemId              itemId;
    Qty                 qty;
    InventBatchId       inventBathcId;
    wmslocationId       ubicacion;
    CrearPicking        crearPicking;
    ;

    select firstonly pesos
        order by Totalpeso;

        PesoMinimo = pesos.TotalPeso;

    //Guradem en aux la combinació del millor camí
    aux = pesos.combinacion;

    while (i <= conlen(aux))
    {
        v = conpeek(aux,i);

        select * from tmparticulosPedido where tmpArticulosPedido.wMSLocationId == v;

        inventBathcId = tmpArticulosPedido.InventBatchId;

        numOp = numOp + 1;

        TablaInforme.InventBatchId = tmpArticulosPedido.InventBatchId;
        TablaInforme.ItemId = tmpArticulosPedido.ItemId;
        TablaInforme.WMSLocationId = tmpArticulosPedido.wMSLocationId;
        TablaInforme.Qty = tmpArticulosPedido.Qty;
        TablaInforme.InventLocationId = tmpArticulosPedido.InventLocationId;
        TablaInforme.OrdenOp = numOp;

        select * from crearPicking where crearPicking.ItemId == TablaInforme.ItemId;

        TablaInforme.InventBatchId = crearPicking.inventBatchId;

        TablaInforme.insert();

        i++;
    }

}
```

10. Annex 2: Algunes proves

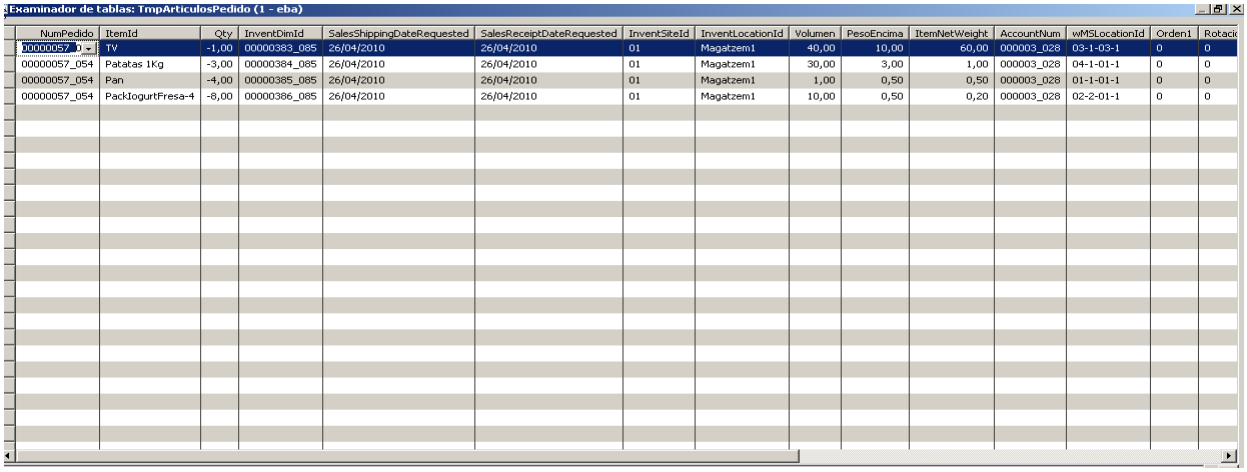
En aquest annex es podrà veure els resultats obtinguts en les diferents taules que utilitza l'algorisme per a processa les dades que va obtenim en les consultes que es van realitzant al llarg del procés. És a partir d'aquestes probes d'on es treuen les conclusions d'aquest projecte.

El primer exemple de comanda de venda és:



Il·lustració 39 Exemple comanda de venta

Els resultats obtinguts en les taules emprades en el procés per al primer magatzem són:



Examinador de tablas: TmpArticulosPedido (1 - eba)

NumPedido	ItemId	Qty	InventDimId	SalesShippingDateRequested	SalesReceiptDateRequested	InventSiteId	InventLocationId	Volumen	PesoEncima	ItemNetWeight	AccountNum	wMSLocationId	Orden1	Rotaci
00000057_0	TV	-1,00	00000383_085	26/04/2010	26/04/2010	01	Magatzem1	40,00	10,00	60,00	000003_028	03-1-03-1	0	0
00000057_054	Patatas 1Kg	-3,00	00000384_085	26/04/2010	26/04/2010	01	Magatzem1	30,00	3,00	1,00	000003_028	04-1-01-1	0	0
00000057_054	Pan	-4,00	00000385_085	26/04/2010	26/04/2010	01	Magatzem1	1,00	0,50	0,50	000003_028	01-1-01-1	0	0
00000057_054	PackIogurtFresa-4	-8,00	00000386_085	26/04/2010	26/04/2010	01	Magatzem1	10,00	0,50	0,20	000003_028	02-2-01-1	0	0

Il·lustració 40 Taula TmpArticulosPedido

Examinador de tablas: TmpPedidosOrdenados (1 - eba)

Qty	InventBatchId	ItemId	InventLocationId	Ordenacion	WMSLocationId	InventDimId1	Dimension	Dimension	Dimension	SalesShippingDateRequested	SalesReceiptDateRequested	AccountNum	InventSiteId	dataAre
-8,00		PackIogurtFresa-4	Magatzem1	3	02-2-01-1	00000386_085				26/04/2010	26/04/2010	000003_028	01	eba
-4,00		Pan	Magatzem1	4	01-1-01-1	00000385_085				26/04/2010	26/04/2010	000003_028	01	eba
-3,00		Patatas 1Kg	Magatzem1	2	04-1-01-1	00000384_085				26/04/2010	26/04/2010	000003_028	01	eba
-1,00		TV	Magatzem1	1	03-1-03-1	00000383_085				26/04/2010	26/04/2010	000003_028	01	eba

Il·lustració 41 Taula TmpPedidosOrdenados

Examinador de tablas: TmpCaducidad (1 - eba)

ItemId	InventBatchId	expDate	SalesShippingDateRequested	SalesReceiptDateRequested	Dimension	Dimension	Dimension	Valoracion	Rotacion	dataAreaId	recVersion	RecId
TV			26/04/2010	26/04/2010				FIFO	40	eba	1	5637151084
Patatas 1Kg		05/06/2010	26/04/2010	26/04/2010				FIFO	40	eba	1	5637151085
PackIogurtFresa-4		05/06/2010	26/04/2010	26/04/2010				FIFO	40	eba	1	5637151086
Pan		05/06/2010	26/04/2010	26/04/2010				FIFO	40	eba	1	5637151087

Il·lustració 42 Taula TmpCaducidad

Examinador de tablas: CrearPicking (1 - eba)

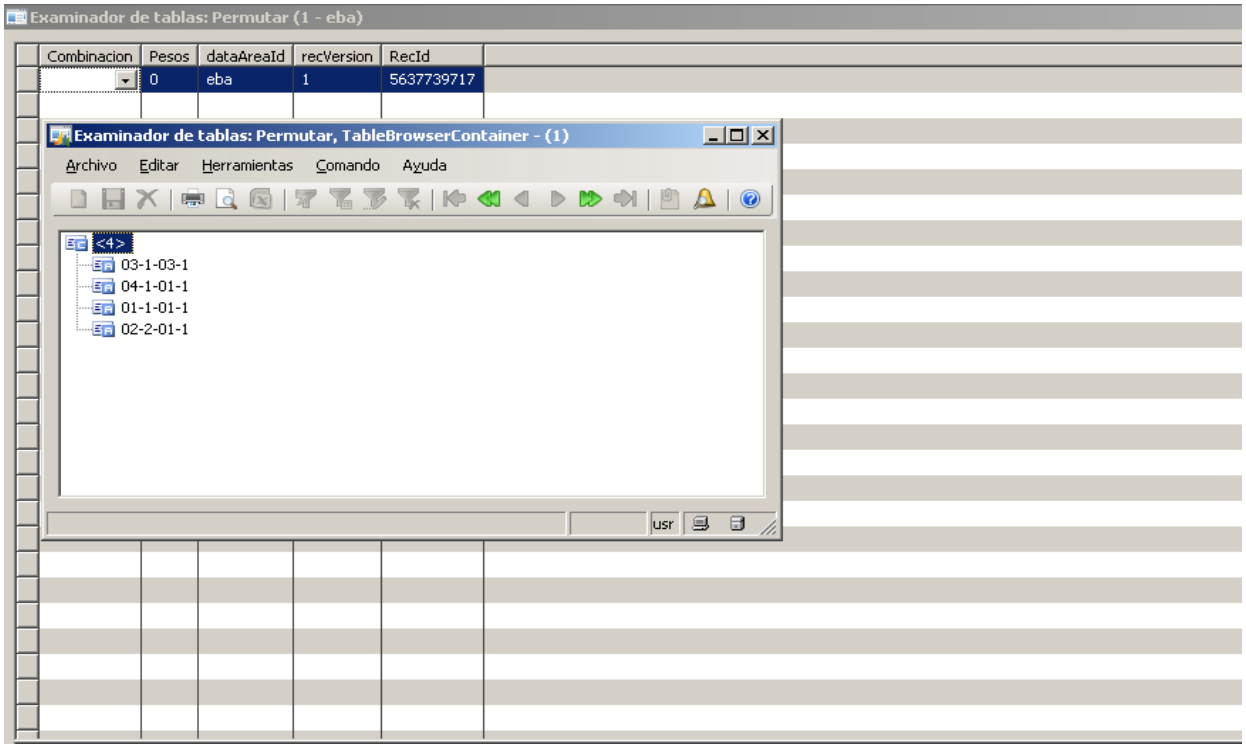
ItemId	InventDimId	WMSLocationId	InventBatchId	Qty	InventLocationId	InventSiteId	dataAreaId	recVersion	RecId
TV		03-1-03-1	00001	-1,00	Magatzem1	01	eba	1	5637149834
Patatas 1Kg		04-1-01-1	00002	-3,00	Magatzem1	01	eba	1	5637149835
PackIogurtFresa-4		02-2-01-1	00004	-8,00	Magatzem1	01	eba	1	5637149836
Pan		01-1-01-1	00003	-4,00	Magatzem1	01	eba	1	5637149837

Il·lustració 43 Taula CrearPicking

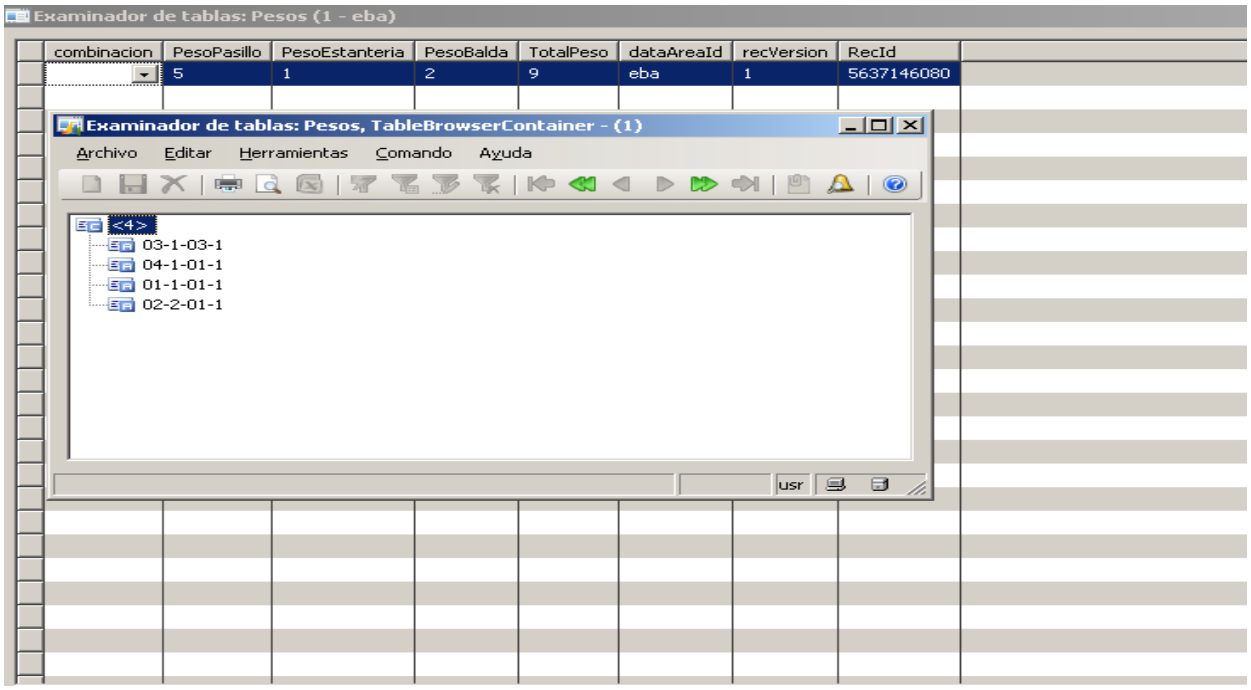
Examinador de tablas: EstocDisponible (1 - eba)

ItemId	InventBatchId	Qty	dataAreaId	recVersion	RecId
TV	00001	116,00	eba	1	5637149596
Patatas 1Kg	00002	118,00	eba	1	5637149597
PackIogurtFresa-4	00004	96,00	eba	1	5637149598
Pan	00003	110,00	eba	1	5637149599

Il·lustració 44 Taula Estocdisponible



Il·lustració 45 Taula Permutar



Il·lustració 46 Taula Pesos

Examinador de tablas: SegOperacions (1 - eba)

ItemId	Qty	InventLocationId	InventSiteId	InventDimId	InventBatchId	OrdenOp	Operacion	InventQty	WMSLocationId	ItemId1	dataAreaId	recVersion	RecId
TV	-1,00	Magatzem1			00001	1	Picking	0,00	03-1-03-1		eba	1	5637563846
Patatas 1Kg	-3,00	Magatzem1			00002	2	Picking	0,00	04-1-01-1		eba	1	5637563847
PackIogurtFresa-4	-8,00	Magatzem1			00004	3	Picking	0,00	02-2-01-1		eba	1	5637563848
Pan	-4,00	Magatzem1			00003	4	Picking	0,00	01-1-01-1		eba	1	5637563849

Il·lustració 47 Taula SegOperacions

Examinador de tablas: TablaInforme (1 - eba)

ItemId	Qty	InventBatchId	InventLocationId	WMSLocationId	OrdenOp	dataAreaId	recVersion	RecId
TV	-1,00	00001	Magatzem1	03-1-03-1	1	eba	1	5637144842
Patatas 1Kg	-3,00	00002	Magatzem1	04-1-01-1	2	eba	1	5637144843
Pan	-4,00	00003	Magatzem1	01-1-01-1	3	eba	1	5637144844
PackIogurtFresa-4	-8,00	00004	Magatzem1	02-2-01-1	4	eba	1	5637144845

Il·lustració 48 Taula TablaInforme

L'informe que s'entregarà a l'usuari amb el resultat de la ruta de Picking és el següent:

Orden Operacion	Código de artículo	Número de lote	Cantidad	Ubicación	Almacén
1	TV	00001	-1,00	03-1-01-1	Magatzem1
2	Patatas 1Kg	00002	-3,00	04-1-01-1	Magatzem1
3	Pan	00003	-4,00	01-1-01-1	Magatzem1
4	PacklogurtFresa-4	00004	-8,00	02-2-01-1	Magatzem1

Il·lustració 49 Llistat de ruta de Picking PV 57_054

Es realitza la mateixa comanda de venta que l'anterior però aquets ara es servirà des del segon magatzem.

Els resultats obtinguts en les taules emprades en el procés són:

Examinador de tablas: TmpArticulosPedido (1 - eba)

NumPedido	ItemId	Qty	InventDimId	SalesShippingDateRequested	SalesReceiptDateRequested	InventSiteId	InventLocationId	Volumen	PesoEncima	ItemNetWeight	AccountNum	wMSLocationId	Orden1	Rotaci
00000058_0	TV	-1,00	00000425_085	07/05/2010	07/05/2010	01	Magatzem2	40,00	10,00	60,00	000001_028	04-2-03-2	0	0
00000058_054	Patatas 1Kg	-3,00	00000427_085	07/05/2010	07/05/2010	01	Magatzem2	30,00	3,00	1,00	000001_028	02-1-01-2	0	0
00000058_054	Pan	-4,00	00000429_085	07/05/2010	07/05/2010	01	Magatzem2	1,00	0,50	0,50	000001_028	04-2-03-1	0	0
00000058_054	PacklogurtFresa-4	-8,00	00000431_085	07/05/2010	07/05/2010	01	Magatzem2	10,00	0,50	0,20	000001_028	03-1-03-1	0	0

Il·lustració 50 Taula TmpArticulosPedido

Examinador de tablas: TmpPedidosOrdenados (1 - eba)

Qty	InventBatchId	ItemId	InventLocationId	Ordenacion	WMSLocationId	InventDimId1	Dimension	Dimension	Dimension	SalesShippingDateRequested	SalesReceiptDateRequested	AccountNum	InventSiteId	dataAre
-8,00		PackIogurtFresa-4	Magatzem2	3	03-1-03-1	00000431_085				07/05/2010	07/05/2010	000001_028	01	eba
-4,00		Pan	Magatzem2	4	04-2-03-1	00000429_085				07/05/2010	07/05/2010	000001_028	01	eba
-3,00		Patatas 1Kg	Magatzem2	2	02-1-01-2	00000427_085				07/05/2010	07/05/2010	000001_028	01	eba
-1,00		TV	Magatzem2	1	04-2-03-2	00000425_085				07/05/2010	07/05/2010	000001_028	01	eba

Il·lustració 51 Taula TmpPedidosOrdenados

Examinador de tablas: TmpCaducidad (1 - eba)

ItemId	InventBatchId	expDate	SalesShippingDateRequested	SalesReceiptDateRequested	Dimension	Dimension	Dimension	Valoracion	Rotacion	dataAreaId	recVersion	RecId
TV			07/05/2010	07/05/2010				LIFO	10	eba	1	5637151156
Patatas 1Kg		17/05/2010	07/05/2010	07/05/2010				LIFO	10	eba	1	5637151157
PackIogurtFresa-4		17/05/2010	07/05/2010	07/05/2010				LIFO	10	eba	1	5637151158
Pan		17/05/2010	07/05/2010	07/05/2010				LIFO	10	eba	1	5637151159

Il·lustració 52 Taula TmpCaducidad

Examinador de tablas: CrearPicking (1 - eba)

ItemId	InventDimId	WMSLocationId	InventBatchId	Qty	InventLocationId	InventSiteId	dataAreaId	recVersion	RecId
TV		04-2-03-2	00011	-1,00	Magatzem2	01	eba	1	5637149906
Patatas 1Kg		02-1-01-2	00022	-3,00	Magatzem2	01	eba	1	5637149907
PackIogurtFresa-4		03-1-03-1	00044	-8,00	Magatzem2	01	eba	1	5637149908
Pan		04-2-03-1	00033	-4,00	Magatzem2	01	eba	1	5637149909

Il·lustració 53 Taula CrearPicking

Examinador de tablas: EstocDisponible (1 - eba)

ItemId	InventBatchId	Qty	dataAreaId	recVersion	RecId
TV	00011	116,00	eba	1	5637149656
Patatas 1Kg	00022	118,00	eba	1	5637149657
PackIogurtFresa-4	00044	96,00	eba	1	5637149658
Pan	00033	110,00	eba	1	5637149659

Il·lustració 54 Taula EstocDisponible

Examinador de tablas: Permutar (1 - eba)

Combinacion	Pesos	dataAreaId	recVersion	RecId
	0	eba	1	5637740101

Il·lustració 55 Taula Permutar

Examinador de tablas: Pesos (1 - eba)

combinacion	PesoPasillo	PesoEstanteria	PesoBalda	TotalPeso	dataAreaId	recVersion	RecId
	5	3	4	26	eba	1	5637146090

Il·lustració 56 Taula Pesos

Examinador de tablas: TablaInforme (1 - eba)

ItemId	Qty	InventBatchId	InventLocationId	WMSLocationId	OrdenOp	dataAreaId	recVersion	RecId
TV	-1,00	00011	Magatzem2	04-2-03-2	1	eba	1	5637144878
Patatas 1Kg	-3,00	00022	Magatzem2	02-1-01-2	2	eba	1	5637144879
Pan	-4,00	00033	Magatzem2	04-2-03-1	3	eba	1	5637144880
PackIogurtFresa-4	-8,00	00044	Magatzem2	03-1-03-1	4	eba	1	5637144881

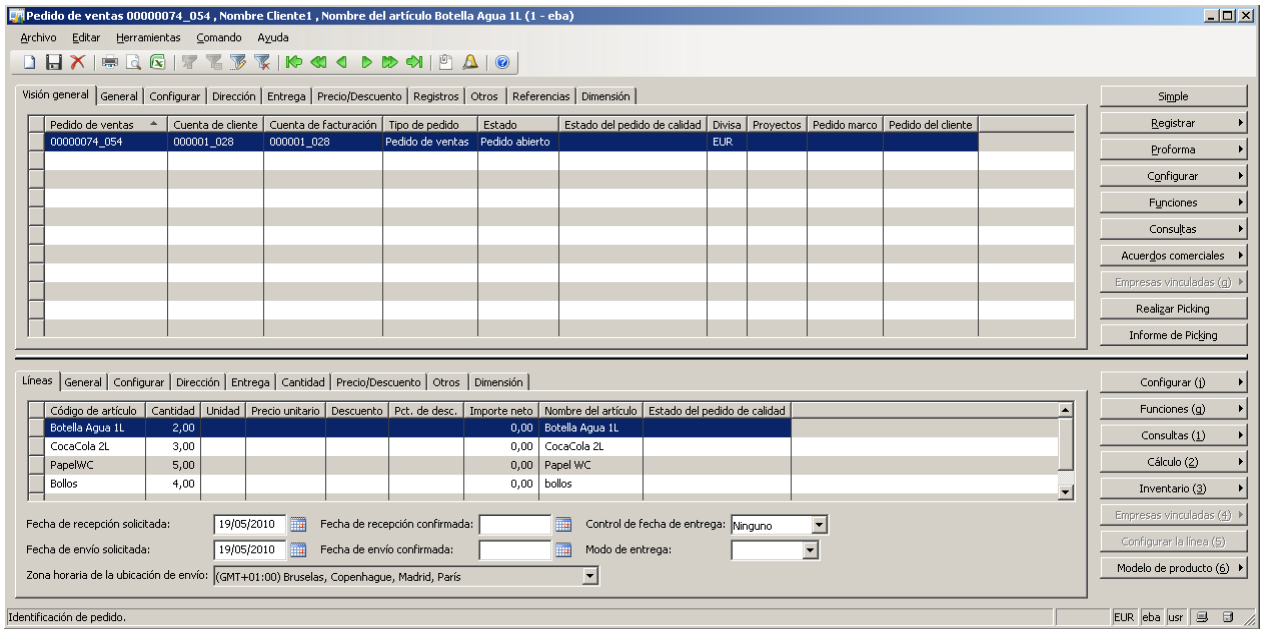
Il·lustració 57 Taula TablaInforme

L'informe que s'entregarà a l'usuari amb el resultat de la ruta de Picking és el següent:

Orden Operacion	Código de artículo	Número de lote	Cantidad	Ubicación	Almacén
1	TV	00011	-1,00	04-2-03-2	Magatzem2
2	Patatas 1Kg	00022	-3,00	02-1-01-2	Magatzem2
3	Pan	00033	-4,00	04-2-03-1	Magatzem2
4	PackIogurtFresa-4	00044	-8,00	03-1-03-1	Magatzem2

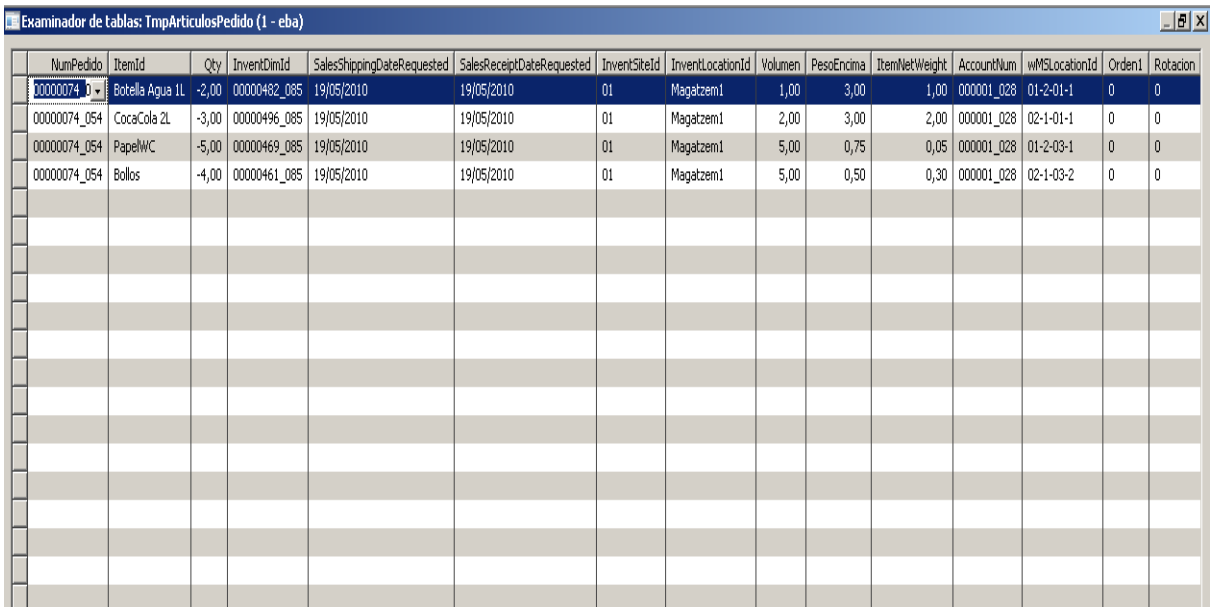
Il·lustració 58 Llistat de ruta de Picking PV 58_054

El segon exemple de comanda de venda és:



Il·lustració 59 Exemple comanda de venta

Els resultats obtinguts en les taules emprades en el procés per al primer magatzem són:



NumPedido	ItemId	Qty	InventDimId	SalesShippingDateRequested	SalesReceiptDateRequested	InventSiteId	InventLocationId	Volumen	PesoEncima	ItemNetWeight	AccountNum	wMSLocationId	Orden1	Rotacion
00000074_0	Botella Agua 1L	-2,00	00000482_085	19/05/2010	19/05/2010	01	Magatzem1	1,00	3,00	1,00	000001_028	01-2-01-1	0	0
00000074_054	CocaCola 2L	-3,00	00000496_085	19/05/2010	19/05/2010	01	Magatzem1	2,00	3,00	2,00	000001_028	02-1-01-1	0	0
00000074_054	PapelWC	-5,00	00000469_085	19/05/2010	19/05/2010	01	Magatzem1	5,00	0,75	0,05	000001_028	01-2-03-1	0	0
00000074_054	Bollos	-4,00	00000461_085	19/05/2010	19/05/2010	01	Magatzem1	5,00	0,50	0,30	000001_028	02-1-03-2	0	0

Il·lustració 60 Taula TmpArticulosPedido

Examinador de tablas: TmpPedidosOrdenados (1 - eba)

Qty	InventBatchId	ItemId	InventLocationId	Ordenacion	WMSLocationId	InventDimId1	Dimension	Dimension	Dimension	SalesShippingDateRequested	SalesReceiptDateRequested	AccountNum	InventSiteId	dataAreaId
-4,00		Bollos	Magatzem1	4	02-1-03-2	00000461_085				19/05/2010	19/05/2010	000001_028	01	eba
-2,00		Botella Agua 1L	Magatzem1	2	01-2-01-1	00000482_085				19/05/2010	19/05/2010	000001_028	01	eba
-3,00		CocaCola 2L	Magatzem1	1	02-1-01-1	00000496_085				19/05/2010	19/05/2010	000001_028	01	eba
-5,00		PapelWC	Magatzem1	3	01-2-03-1	00000469_085				19/05/2010	19/05/2010	000001_028	01	eba

Il·lustració 61 Taula TmpPedidosOrdenados

Examinador de tablas: TmpCaducidad (1 - eba)

ItemId	InventBatchId	expDate	SalesShippingDateRequested	SalesReceiptDateRequested	Dimension	Dimension	Dimension	Valoracion	Rotacion	dataAreaId	recVersion	RecId
CocaCola 2L	00066	29/05/2010	19/05/2010	19/05/2010				LIFO	10	eba	1	5637152993
Botella Agua 1L	00077	29/05/2010	19/05/2010	19/05/2010				LIFO	10	eba	1	5637152994
PapelWC	114	29/05/2010	19/05/2010	19/05/2010				LIFO	10	eba	1	5637152995
Bollos	11114	29/05/2010	19/05/2010	19/05/2010				LIFO	10	eba	1	5637152996

Il·lustració 62 Taula TmpCaducidad

Examinador de tablas: CrearPicking (1 - eba)

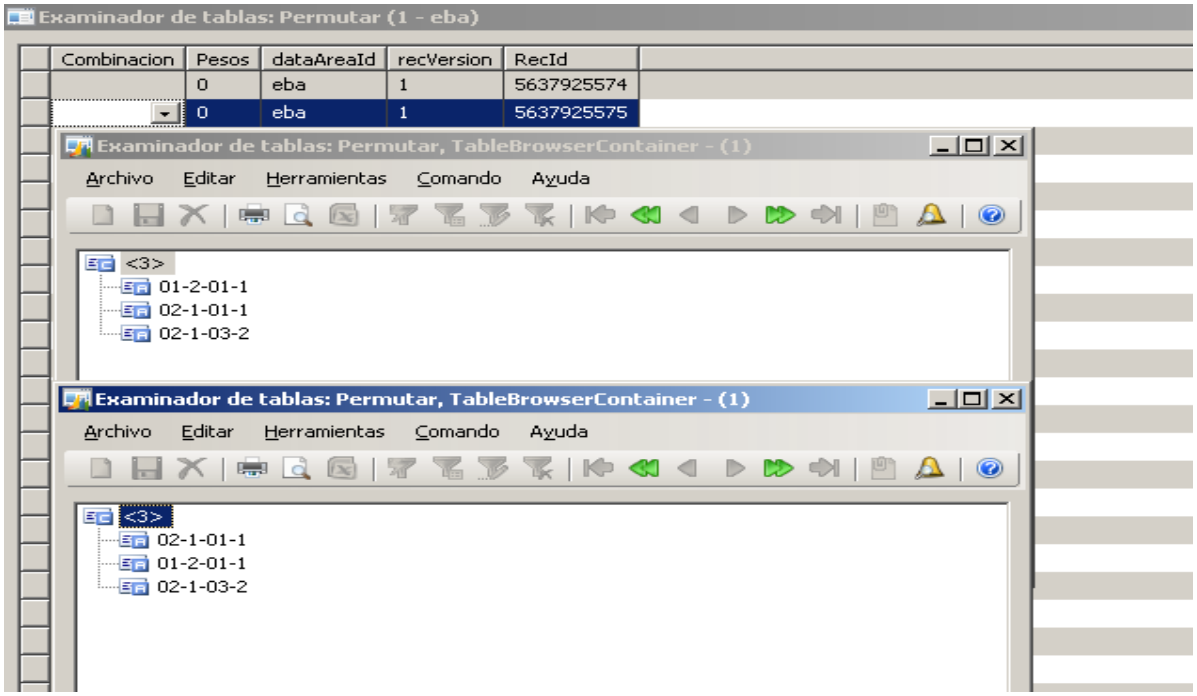
ItemId	InventDimId	WMSLocationId	InventBatchId	Qty	InventLocationId	InventSiteId	dataAreaId	recVersion	RecId
CocaCola 2L	00000304_085	02-1-01-1	00066	-3,00	Magatzem1	01	eba	1	5637151743
Botella Agua 1L	00000306_085	01-2-01-1	00077	-2,00	Magatzem1	01	eba	1	5637151744
PapelWC		01-2-03-1	114	-5,00	Magatzem1	01	eba	1	5637151745
Bollos	00000489_085	02-1-03-2	11114	-4,00	Magatzem1	01	eba	1	5637151746

Il·lustració 63 Taula CrearPicking

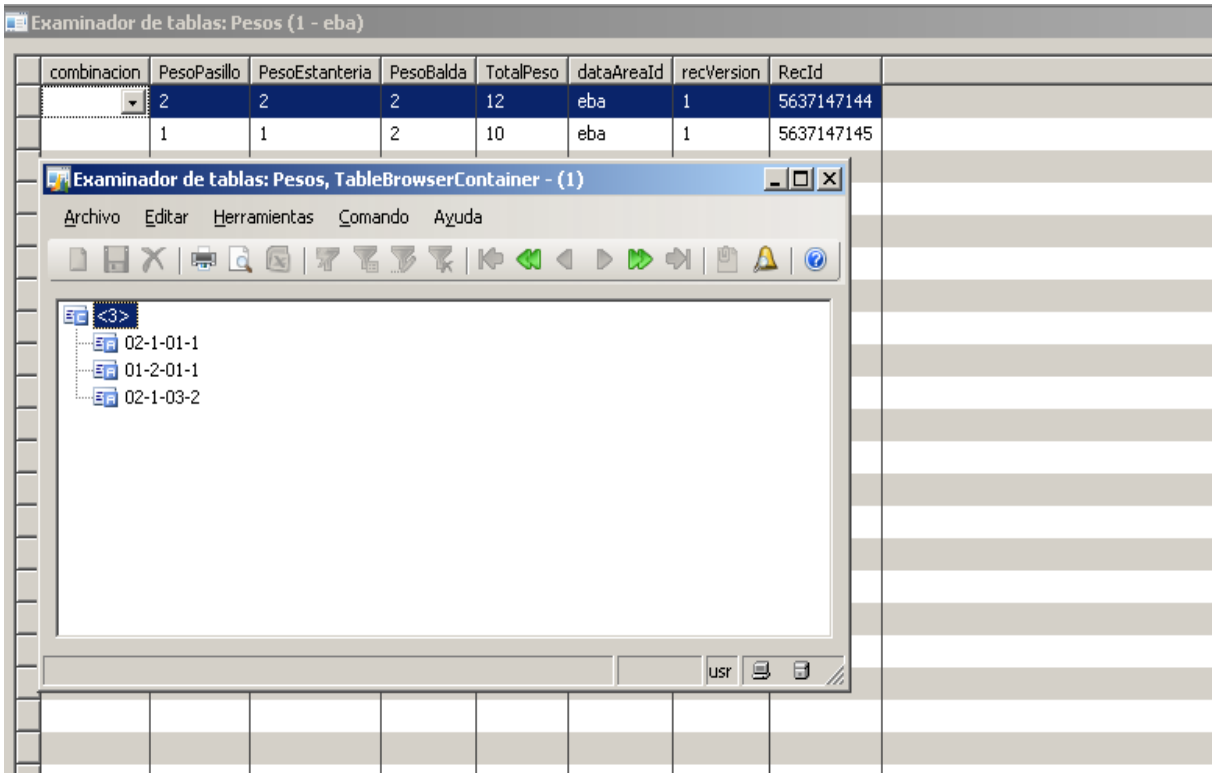
Examinador de tablas: EstocDisponible (1 - eba)

ItemId	InventBatchId	Qty	dataAreaId	recVersion	RecId
CocaCola 2L	00066	17,00	eba	1	5637151428
Botella Agua 1L	00077	18,00	eba	1	5637151429
Bollos	11114	6,00	eba	1	5637151430

Il·lustració 64 Taula EstocDisponible



Il·lustració 65 Taula Permutar



Il·lustració 66 Taula Pesos

Examinador de tablas: Pesos (1 - eba)

combinacion	PesoPasillo	PesoEstanteria	PesoBalda	TotalPeso	dataAreaId	recVersion	RecId
	2	2	2	12	eba	1	5637147144
	1	1	2	10	eba	1	5637147145

Examinador de tablas: Pesos, TableBrowserContainer - (1)

Archivo Editar Herramientas Comando Ayuda

<3>
01-2-01-1
02-1-01-1
02-1-03-2

usr

Il·lustració 67 Taula Pesos

Examinador de tablas: SegOperacions (1 - eba)

ItemId	Qty	InventLocationId	InventSiteId	InventDimId	InventBatchId	OrdenOp	Operacion	InventQty	WMSLocationId	ItemId1	dataAreaId	recVersion	RecId
CocaCola 2L	-3,00	Magatzem1		00000304_085	00066	1	Picking	0,00	02-1-01-1		eba	1	5637565743
Botella Agua 1L	-2,00	Magatzem1		00000306_085	00077	2	Picking	0,00	01-2-01-1		eba	1	5637565744
PapelWC	5,00	Magatzem1				3	No Operacion	0,00			eba	1	5637565745
Bollos	-4,00	Magatzem1		00000489_085	11114	4	Picking	0,00	02-1-03-2		eba	1	5637565746

Il·lustració 68 Taula SegOperacions

Examinador de tablas: TablaInforme (1 - eba)

ItemId	Qty	InventBatchId	InventLocationId	WMSLocationId	OrdenOp	dataAreaId	recVersion	RecId
Botella Agua	-2,00	00077	Magatzem1	01-2-01-1	1	eba	1	5637145968
CocaCola 2L	-3,00	00066	Magatzem1	02-1-01-1	2	eba	1	5637145969
Bollos	-4,00	11114	Magatzem1	02-1-03-2	3	eba	1	5637145970

Il·lustració 69 Taula TablaInforme

L'informe que s'entregarà a l'usuari amb el resultat de la ruta de Picking és el següent:

Orden Operacion	Código de artículo	Número de lote	Cantidad	Ubicación	Almacén
1	Botella Agua 1L	00077	-2,00	01-2-01-1	Magatzem1
2	CocaCola 2L	00066	-3,00	02-1-01-1	Magatzem1
3	Bollos	11114	-4,00	02-1-03-2	Magatzem1

Il·lustració 70 Llistat de ruta de Picking PV 74_054

Es realitza la mateixa comanda de venta que l'anterior però aquets ara es servirà des del segon magatzem.

Els resultats obtinguts en les taules emprades en el procés són:

Examinador de tablas: TmpArticulosPedido (1 - eba)

NumPedido	ItemId	Qty	InventDimId	SalesShippingDateRequested	SalesReceiptDateRequested	InventSiteId	InventLocationId	Volumen	PesoEncima	ItemNetWeight	AccountNum	wMSLocationId	Orden1	Rotacion
00000075_	Botella Agua 1L	-2,00	00000485_085	19/05/2010	19/05/2010	01	Magatzem2	1,00	3,00	1,00	000001_028	01-2-01-1	0	0
00000075_054	CocaCola 2L	-3,00	00000497_085	19/05/2010	19/05/2010	01	Magatzem2	2,00	3,00	2,00	000001_028	02-1-01-1	0	0
00000075_054	PapelWC	-5,00	00000498_085	19/05/2010	19/05/2010	01	Magatzem2	5,00	0,75	0,05	000001_028	03-2-01-1	0	0
00000075_054	Bollos	-4,00	00000441_085	19/05/2010	19/05/2010	01	Magatzem2	5,00	0,50	0,30	000001_028	04-2-01-1	0	0

Il·lustració 71 Taula TmpArticulosPedido

Examinador de tablas: TmpPedidosOrdenados (1 - eba)

Qty	InventBatchId	ItemId	InventLocationId	Ordenacion	WMSLocationId	InventDimId1	Dimension	Dimension	Dimension	SalesShippingDateRequested	SalesReceiptDateRequested	AccountNum	InventSiteId	dataAreaId
-4,00		Bollos	Magatzem2	4	04-2-01-1	00000441_085				19/05/2010	19/05/2010	000001_028	01	eba
-2,00		Botella Agua 1L	Magatzem2	2	01-2-01-1	00000485_085				19/05/2010	19/05/2010	000001_028	01	eba
-3,00		CocaCola 2L	Magatzem2	1	02-1-01-1	00000497_085				19/05/2010	19/05/2010	000001_028	01	eba
-5,00		PapelWC	Magatzem2	3	03-2-01-1	00000498_085				19/05/2010	19/05/2010	000001_028	01	eba

Il·lustració 72 Taula tmpPedidosOrdenados

Examinador de tablas: TmpCaducidad (1 - eba)

ItemId	InventBatchId	expDate	SalesShippingDateRequested	SalesReceiptDateRequested	Dimension	Dimension	Dimension	Valoracion	Rotacion	dataAreaId	recVersion	RecId
CocaCola 2L	00066	29/05/2010	19/05/2010	19/05/2010				LIFO	10	eba	1	5637152997
Botella Agua 1L	00077	29/05/2010	19/05/2010	19/05/2010				LIFO	10	eba	1	5637152998
PapelWC	114	29/05/2010	19/05/2010	19/05/2010				LIFO	10	eba	1	5637152999
Bollos	11114	29/05/2010	19/05/2010	19/05/2010				LIFO	10	eba	1	5637153000

Il·lustració 73 Taula tmpCaducidad

Examinador de tablas: CrearPicking (1 - eba)

ItemId	InventDimId	WMSLocationId	InventBatchId	Qty	InventLocationId	InventSiteId	dataAreaId	recVersion	RecId
CocaCola 2L		02-1-01-1	00066	-3,00	Magatzem2	01	eba	1	5637151747
Botella Agua 1L		01-2-01-1	00077	-2,00	Magatzem2	01	eba	1	5637151748
PapelWC		03-2-01-1	114	-5,00	Magatzem2	01	eba	1	5637151749
Bollos	00000490_085	04-2-01-1	11114	-4,00	Magatzem2	01	eba	1	5637151750

Il·lustració 74 Taula CrearPicking

Examinador de tablas: EstocDisponible (1 - eba)

ItemId	InventBatchId	Qty	dataAreaId	recVersion	RecId
CocaCola 2L	00066	70,00	eba	1	5637151431
Botella Agua 1L	00077	67,00	eba	1	5637151432
Bollos	11114	6,00	eba	1	5637151433

Il·lustració 75 Taula EstocDisponible

Examinador de tablas: Permutar (1 - eba)

Combinacion	Pesos	dataAreaId	recVersion	RecId
	0	eba	1	5637925580
	0	eba	1	5637925581

Examinador de tablas: Permutar, TableBrowserContainer - (1)

Archivo Editar Herramientas Comando Ayuda

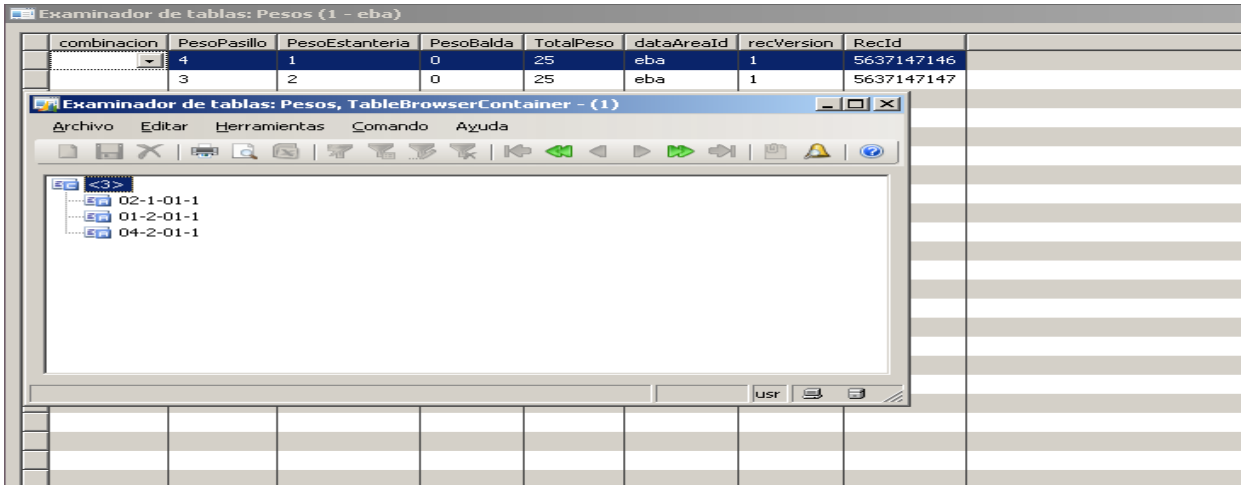
<3>
02-1-01-1
01-2-01-1
04-2-01-1

Examinador de tablas: Permutar, TableBrowserContainer - (1)

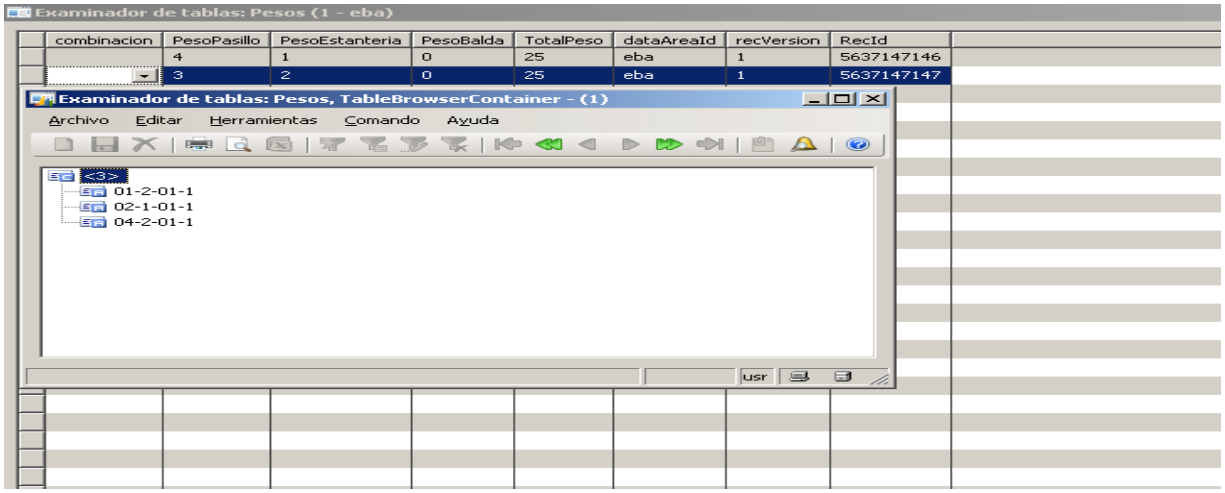
Archivo Editar Herramientas Comando Ayuda

<3>
01-2-01-1
02-1-01-1
04-2-01-1

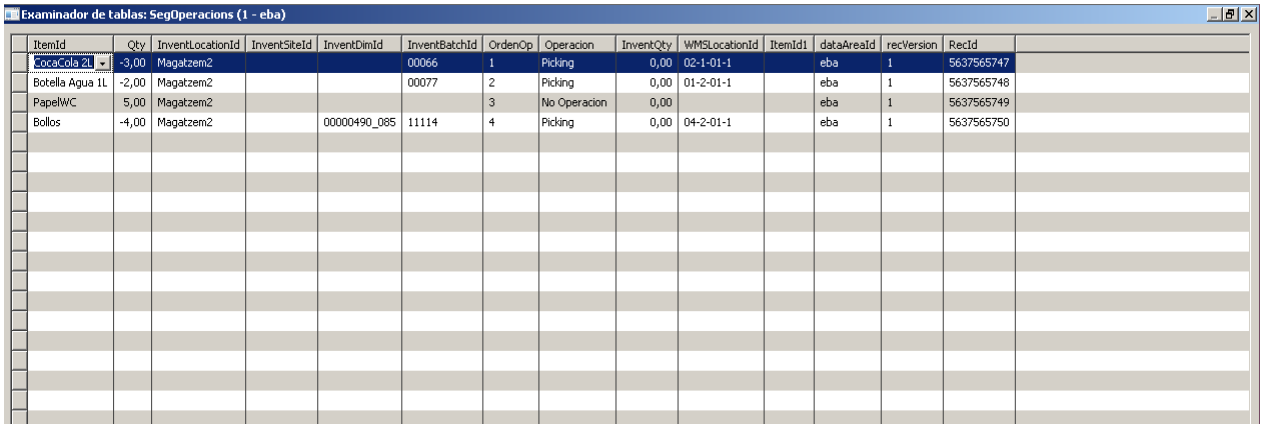
Il·lustració 76 Taula Permutar



Il·lustració 77 Taula Pesos



Il·lustració 78 Taula Pesos



ItemId	Qty	InventLocationId	InventSiteId	InventDimId	InventBatchId	OrdenOp	Operacion	InventQty	WMSLocationId	ItemId1	dataAreaId	recVersion	RecId
CocaCola 2L	-3,00	Magatzem2			00066	1	Picking	0,00	02-1-01-1		eba	1	5637565747
Botella Agua 1L	-2,00	Magatzem2			00077	2	Picking	0,00	01-2-01-1		eba	1	5637565748
PapelWC	5,00	Magatzem2				3	No Operacion	0,00			eba	1	5637565749
Bollos	-4,00	Magatzem2		00000490_085	11114	4	Picking	0,00	04-2-01-1		eba	1	5637565750

Il·lustració 79 Taula SegOperacions

Examinador de tablas: TablaInforme (1 - eba)

ItemId	Qty	InventBatchId	InventLocationId	WMSLocationId	OrdenOp	dataAreaId	recVersion	RecId
Botella Agua	-2,00	00077	Magatzem2	01-2-01-1	1	eba	1	5637145971
CocaCola 2L	-3,00	00066	Magatzem2	02-1-01-1	2	eba	1	5637145972
Bollos	-4,00	11114	Magatzem2	04-2-01-1	3	eba	1	5637145973

Il·lustració 80 Taula TablaInforme

L'informe que s'entregarà a l'usuari amb el resultat de la ruta de Picking és el següent:

Orden Operacion	Código de artículo	Número de lote	Cantidad	Ubicación	Almacén
1	Botella Agua 1L	00077	-2,00	01-2-01-1	Magatzem2
2	CocaCola 2L	00066	-3,00	02-1-01-1	Magatzem2
3	Bollos	11114	-4,00	04-2-01-1	Magatzem2

Il·lustració 81 Llistat de ruta de Picking PV 75_054

El tercer exemple de comanda de venda és:

Pedido de ventas 00000076_054 , Nombre Cliente2 , Nombre del artículo Paxk Leche 12 ud (1 - eba)

Archivo Editar Herramientas Comando Ayuda

Visión general | General | Configurar | Dirección | Entrega | Precio/Descuento | Registros | Otros | Referencias | Dimensión

Pedido de ventas	Cuenta de cliente	Cuenta de facturación	Tipo de pedido	Estado	Estado del pedido de calidad	Divisa	Proyectos	Pedido marco	Pedido del cliente
00000076_054	000002_028	000002_028	Pedido de ventas	Pedido abierto		EUR			

Simple
Registrar
Proforma
Configurar
Funciones
Consultas
Acuerdos comerciales
Empresas vinculadas (g)
Realizar Picking
Informe de Picking

Líneas | General | Configurar | Dirección | Entrega | Cantidad | Precio/Descuento | Otros | Dimensión

Código de artículo	Cantidad	Unidad	Precio unitario	Descuento	Pct. de desc.	Importe neto	Nombre del artículo	Estado del pedido de calidad
PackLeche12ud	1,00					0,00	Paxk Leche 12 ud	
Galletas	5,00					0,00	Galletas	
PapelWC	3,00					0,00	Papel WC	
Jabon	2,00					0,00	Jabon	
CocaCola 2L	3,00					0,00	CocaCola 2L	

Configurar (i)
Funciones (g)
Consultas (1)
Cálculo (2)
Inventario (3)
Empresas vinculadas (4)
Configurar la línea (5)
Modelo de producto (6)

Fecha de recepción solicitada: 20/05/2010 Fecha de recepción confirmada: Control de fecha de entrega: Ninguno
Fecha de envío solicitada: 20/05/2010 Fecha de envío confirmada: Modo de entrega:
Zona horaria de la ubicación de envío: (GMT+01:00) Bruselas, Copenhague, Madrid, París

dentificación de pedido. EUR eba usr

Il·lustració 82 Exemple comanda de venta

Els resultats obtinguts en les taules emprades en el procés per al primer magatzem són:

Examinador de tablas: TmpArticulosPedido (1 - eba)

NumPedido	ItemId	Qty	InventDimId	SalesShippingDateRequested	SalesReceiptDateRequested	InventSiteId	InventLocationId	Volumen	PesoEncima	ItemNetWeight	AccountNum	wMSLocationId	Orden1	Rotacion
00000076_	PackLeche12ud	-1,00	000000481_085	20/05/2010	20/05/2010	01	Magatzem1	5,00	5,00	12,00	000002_028	01-2-01-2	0	0
00000076_054	Galletas	-5,00	00000491_085	20/05/2010	20/05/2010	01	Magatzem1	0,70	0,50	0,30	000002_028	03-1-01-2	0	0
00000076_054	PapelWC	-3,00	00000469_085	20/05/2010	20/05/2010	01	Magatzem1	5,00	0,75	0,05	000002_028	01-2-03-1	0	0
00000076_054	Jabon	-2,00	00000500_085	20/05/2010	20/05/2010	01	Magatzem1	1,20	2,50	1,00	000002_028	02-2-01-2	0	0
00000076_054	CocaCola 2L	-3,00	00000462_085	20/05/2010	20/05/2010	01	Magatzem1	2,00	3,00	2,00	000002_028	02-1-01-1	0	0
00000076_054	Pan	-1,00	00000504_085	20/05/2010	20/05/2010	01	Magatzem1	1,00	0,50	0,50	000002_028	01-1-01-1	0	0

Il·lustració 83 Taula TmpArticulosPedido

Examinador de tablas: TmpPedidosOrdenados (1 - eba)

Qty	InventBatchId	ItemId	InventLocationId	Ordenacion	WMSLocationId	InventDimId1	Dimension	Dimension	Dimension	SalesShippingDateRequested	SalesReceiptDateRequested	AccountNum	InventSiteId	dataAreaId
-3,00		CocaCola 2L	Magatzem1	2	02-1-01-1	00000462_085				20/05/2010	20/05/2010	000002_028	01	eba
-5,00		Galletas	Magatzem1	6	03-1-01-2	00000491_085				20/05/2010	20/05/2010	000002_028	01	eba
-2,00		Jabon	Magatzem1	3	02-2-01-2	00000500_085				20/05/2010	20/05/2010	000002_028	01	eba
-1,00		PackLeche12ud	Magatzem1	1	01-2-01-2	00000481_085				20/05/2010	20/05/2010	000002_028	01	eba
-1,00		Pan	Magatzem1	5	01-1-01-1	00000504_085				20/05/2010	20/05/2010	000002_028	01	eba
-3,00		PapelWC	Magatzem1	4	01-2-03-1	00000469_085				20/05/2010	20/05/2010	000002_028	01	eba

Il·lustració 84 Taula TmpPedidosOrdenados

Examinador de tablas: TmpCaducidad (1 - eba)

ItemId	InventBatchId	expDate	SalesShippingDateRequested	SalesReceiptDateRequested	Dimension	Dimension	Dimension	Valoracion	Rotacion	dataAreaId	recVersion	RecId
PackLeche12	696	09/06/2010	20/05/2010	20/05/2010				FIFO	20	eba	1	5637153031
CocaCola 2L	01235	09/06/2010	20/05/2010	20/05/2010				FIFO	20	eba	1	5637153032
Jabon	558	09/06/2010	20/05/2010	20/05/2010				FIFO	20	eba	1	5637153033
PapelWC	114	09/06/2010	20/05/2010	20/05/2010				FIFO	20	eba	1	5637153034
Pan	00003	09/06/2010	20/05/2010	20/05/2010				FIFO	20	eba	1	5637153035
Galletas	1114	09/06/2010	20/05/2010	20/05/2010				FIFO	20	eba	1	5637153036

Il·lustració 85 Taula TmpCaducidad

Examinador de tablas: CrearPicking (1 - eba)

ItemId	InventDimId	WMSLocationId	InventBatchId	Qty	InventLocationId	InventSiteId	dataAreaId	recVersion	RecId
PackLeche12		01-2-01-2	696	-1,00	Magatzem1	01	eba	1	5637151781
CocaCola 2L		02-1-01-1	01235	-3,00	Magatzem1	01	eba	1	5637151782
Jabon		02-2-01-2	558	-2,00	Magatzem1	01	eba	1	5637151783
PapelWC		01-2-03-1	114	-3,00	Magatzem1	01	eba	1	5637151784
Pan		01-1-01-1	00003	-1,00	Magatzem1	01	eba	1	5637151785
Galletas		03-1-01-2	1114	-5,00	Magatzem1	01	eba	1	5637151786

Il·lustració 86 Taula CrearPicking

Examinador de tablas: EstocDisponible (1 - eba)

ItemId	InventBatchId	Qty	dataAreaId	recVersion	RecId
PackLeche1	696	59,00	eba	1	5637151444
CocaCola 2L	01235	70,00	eba	1	5637151445
PapelWC	114	197,00	eba	1	5637151446
Pan	00003	113,00	eba	1	5637151447

Il·lustració 87 Taula EstocDisponible

Examinador de tablas: Permutar (1 - eba)

Combinacion	Pesos	dataAreaId	recVersion	RecId
	0	eba	1	5637925599
	0	eba	1	5637925620

Examinador de tablas: Permutar, TableBrowserContainer - (1)

Archivo Editar Herramientas Comando Ayuda

<4>
01-2-01-2
02-1-01-1
01-1-01-1
01-2-03-1

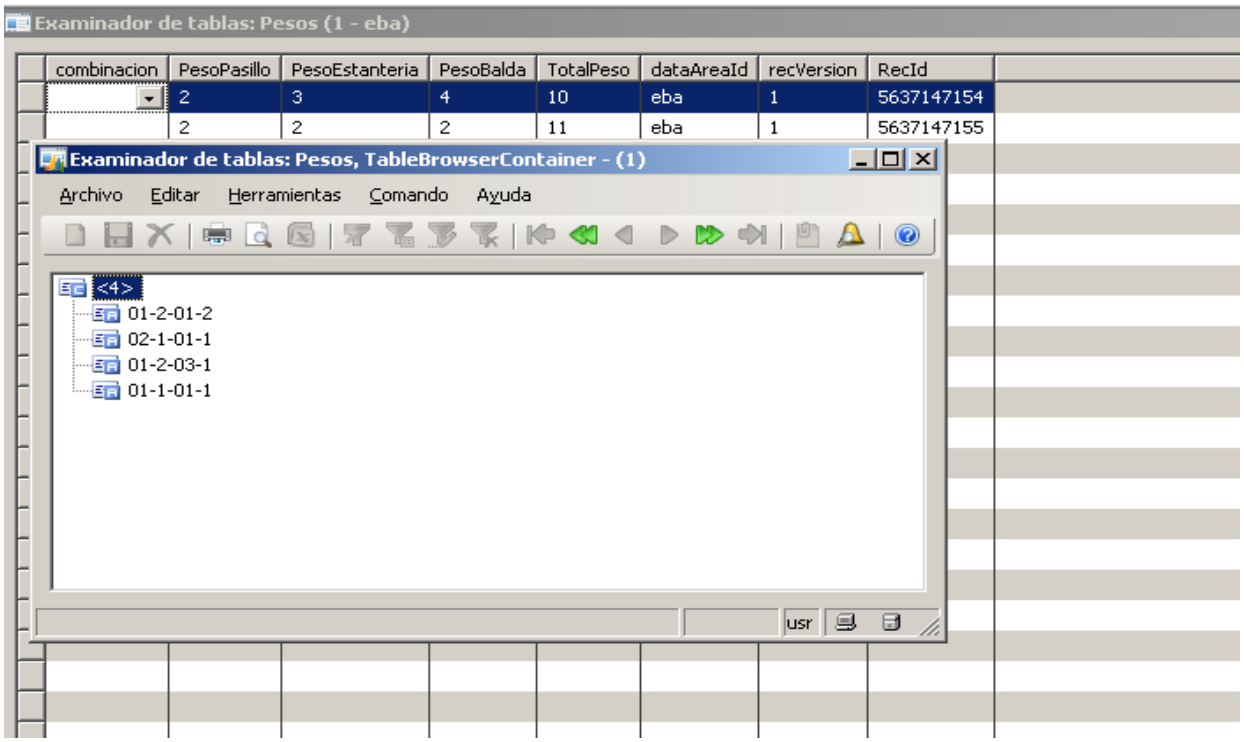
Examinador de tablas: Permutar, TableBrowserContainer - (1)

Archivo Editar Herramientas Comando Ayuda

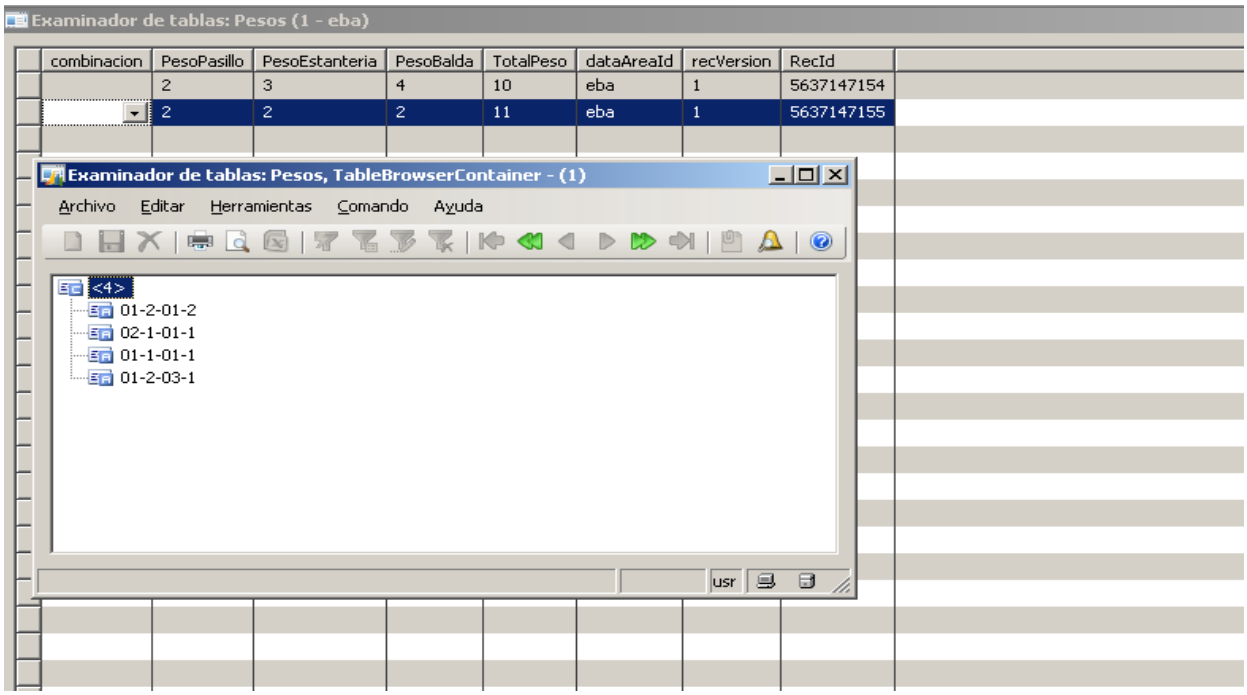
<4>
01-2-01-2
02-1-01-1
01-2-03-1
01-1-01-1

usr

Il·lustració 88 Taula Permutar



Il·lustració 89 Taula Pesos



Il·lustració 90 Taula Pesos

Examinador de tablas: SegOperacions (1 - eba)

ItemId	Qty	InventLocationId	InventSiteId	InventDimId	InventBatchId	OrdenOp	Operacion	InventQty	WMSLocationId	ItemId1	dataAreaId	recVersion	RecId
PackLeche12	-1,00	Magatzem1			696	1	Picking	0,00	01-2-01-2		eba	1	5637565781
CocaCola 2L	-3,00	Magatzem1			01235	2	Picking	0,00	02-1-01-1		eba	1	5637565782
Jabon	2,00	Magatzem1				3	No Operacion	0,00			eba	1	5637565783
PapelWC	-3,00	Magatzem1			114	4	Picking	0,00	01-2-03-1		eba	1	5637565784
Pan	-1,00	Magatzem1			00003	5	Picking	0,00	01-1-01-1		eba	1	5637565785
Galletas	5,00	Magatzem1				6	No Operacion	0,00			eba	1	5637565786

Il·lustració 91 Taula SegOperacions

Examinador de tablas: TablaInforme (1 - eba)

ItemId	Qty	InventBatchId	InventLocationId	WMSLocationId	OrdenOp	dataAreaId	recVersion	RecId
PackLeche12	-1,00	696	Magatzem1	01-2-01-2	1	eba	1	5637145984
CocaCola 2L	-3,00	01235	Magatzem1	02-1-01-1	2	eba	1	5637145985
PapelWC	-3,00	114	Magatzem1	01-2-03-1	3	eba	1	5637145986
Pan	-1,00	00003	Magatzem1	01-1-01-1	4	eba	1	5637145987

Il·lustració 92 Taula TaulaInforme

L'informe que s'entregarà a l'usuari amb el resultat de la ruta de Picking és el següent:

Orden Operacion	Código de artículo	Número de lote	Cantidad	Ubicación	Almacén
1	PackLeche12ud	696	-1,00	01-2-01-2	Magatzem1
2	CocaCola 2L	01235	-3,00	02-1-01-1	Magatzem1
3	PapelWC	114	-3,00	01-2-03-1	Magatzem1
4	Pan	00003	-1,00	01-1-01-1	Magatzem1

Il·lustració 93 Llistat de ruta de Picking PV 76_054

Es realitza la mateixa comanda de venta que l'anterior però aquets ara es servirà des del segon magatzem.

Els resultats obtinguts en les taules emprades en el procés són:

Examinador de tablas: TmpArticulosPedido (1 - eba)

NumPedido	ItemId	Qty	InventDimId	SalesShippingDateRequested	SalesReceiptDateRequested	InventSiteId	InventLocationId	Volumen	PesoEncima	ItemNetWeight	AccountNum	wMSLocationId	Orden1	Rotacion
00000077_0	PackLeche12ud	-1,00	00000505_085	20/05/2010	20/05/2010	01	Magatzem2	5,00	5,00	12,00	000002_028	01-2-01-2	0	0
00000077_054	Galletas	-5,00	00000506_085	20/05/2010	20/05/2010	01	Magatzem2	0,70	0,50	0,30	000002_028	04-1-03-1	0	0
00000077_054	PapelWC	-3,00	00000498_085	20/05/2010	20/05/2010	01	Magatzem2	5,00	0,75	0,05	000002_028	03-2-01-1	0	0
00000077_054	Jabon	-2,00	00000507_085	20/05/2010	20/05/2010	01	Magatzem2	1,20	2,50	1,00	000002_028	02-1-03-2	0	0
00000077_054	CocaCola 2L	-3,00	00000464_085	20/05/2010	20/05/2010	01	Magatzem2	2,00	3,00	2,00	000002_028	02-1-01-1	0	0
00000077_054	Pan	-1,00	00000508_085	20/05/2010	20/05/2010	01	Magatzem2	1,00	0,50	0,50	000002_028	04-1-01-2	0	0

Il·lustració 94 Taula TmpArticulosPedido

Examinador de tablas: TmpPedidosOrdenados (1 - eba)

Qty	InventBatchId	ItemId	InventLocationId	Ordenacion	WMSLocationId	InventDimId1	Dimension	Dimension	Dimension	SalesShippingDateRequested	SalesReceiptDateRequested	AccountNum	InventSiteId	dataAreaId
-3,00		CocaCola 2L	Magatzem2	2	02-1-01-1	00000464_085				20/05/2010	20/05/2010	000002_028	01	eba
-5,00		Galletas	Magatzem2	6	04-1-03-1	00000506_085				20/05/2010	20/05/2010	000002_028	01	eba
-2,00		Jabon	Magatzem2	3	02-1-03-2	00000507_085				20/05/2010	20/05/2010	000002_028	01	eba
-1,00		PackLeche12ud	Magatzem2	1	01-2-01-2	00000505_085				20/05/2010	20/05/2010	000002_028	01	eba
-1,00		Pan	Magatzem2	5	04-1-01-2	00000508_085				20/05/2010	20/05/2010	000002_028	01	eba
-3,00		PapelWC	Magatzem2	4	03-2-01-1	00000498_085				20/05/2010	20/05/2010	000002_028	01	eba

Il·lustració 95 Taula TmpPedidosOrdenados

Examinador de tablas: TmpCaducidad (1 - eba)

ItemId	InventBatchId	expDate	SalesShippingDateRequested	SalesReceiptDateRequested	Dimension	Dimension	Dimension	Valoracion	Rotacion	dataAreaId	recVersion	RecId
PackLeche12	696	09/06/2010	20/05/2010	20/05/2010				FIFO	20	eba	1	5637153037
CocaCola 2L	01235	09/06/2010	20/05/2010	20/05/2010				FIFO	20	eba	1	5637153038
Jabon	558	09/06/2010	20/05/2010	20/05/2010				FIFO	20	eba	1	5637153039
PapelWC	114	09/06/2010	20/05/2010	20/05/2010				FIFO	20	eba	1	5637153040
Pan	00003	09/06/2010	20/05/2010	20/05/2010				FIFO	20	eba	1	5637153041
Galletas	1114	09/06/2010	20/05/2010	20/05/2010				FIFO	20	eba	1	5637153042

Il·lustració 96 Taula TmpCaducidad

Examinador de tablas: CrearPicking (1 - eba)

ItemId	InventDimId	WMSLocationId	InventBatchId	Qty	InventLocationId	InventSiteId	dataAreaId	recVersion	RecId
PackLeche12		01-2-01-2	696	-1,00	Magatzem2	01	eba	1	5637151787
CocaCola 2L		02-1-01-1	01235	-3,00	Magatzem2	01	eba	1	5637151788
Jabon		02-1-03-2	558	-2,00	Magatzem2	01	eba	1	5637151789
PapelWC		03-2-01-1	114	-3,00	Magatzem2	01	eba	1	5637151790
Pan		04-1-01-2	00003	-1,00	Magatzem2	01	eba	1	5637151791
Galletas		04-1-03-1	1114	-5,00	Magatzem2	01	eba	1	5637151792

Il·lustració 97 Taula CrearPicking

Examinador de tablas: EstocDisponible (1 - eba)

ItemId	InventBatchId	Qty	dataAreaId	recVersion	RecId
PackLeche12	696	59,00	eba	1	5637151448
CocaCola 2L	01235	70,00	eba	1	5637151449
PapelWC	114	197,00	eba	1	5637151450
Pan	00003	113,00	eba	1	5637151451

Il·lustració 98 Taula EstocDisponible

Examinador de tablas: Permutar (1 - eba)

Combinacion	Pesos	dataAreaId	recVersion	RecId
	0	eba	1	5637925623
	0	eba	1	5637925644

Examinador de tablas: Permutar, TableBrowserContainer - (1)

Archivo Editar Herramientas Comando Ayuda

<4>
- 01-2-01-2
- 02-1-01-1
- 04-1-01-2
- 03-2-01-1

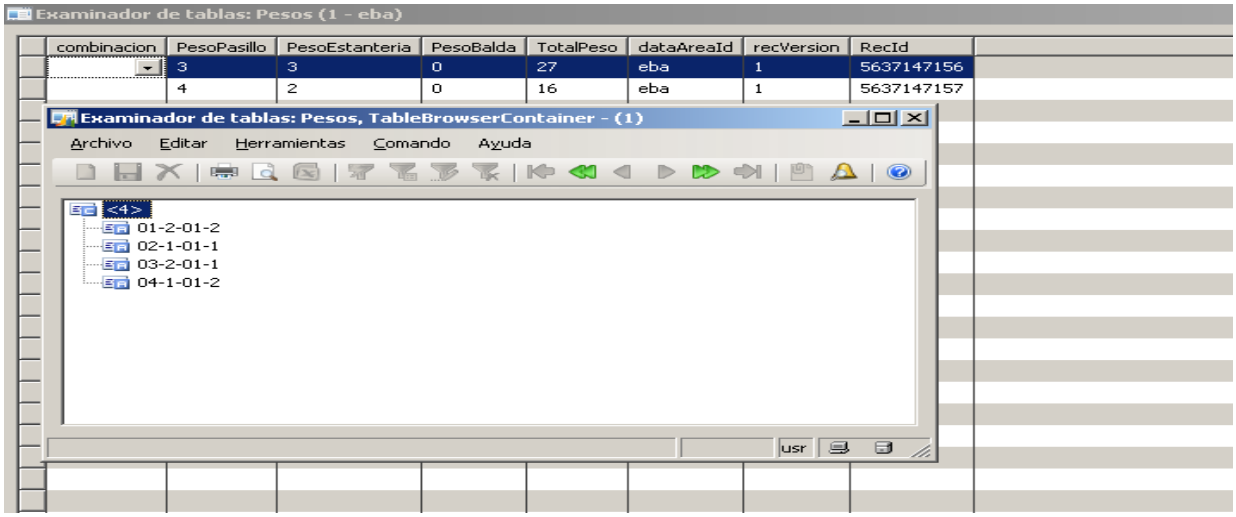
Examinador de tablas: Permutar, TableBrowserContainer - (1)

Archivo Editar Herramientas Comando Ayuda

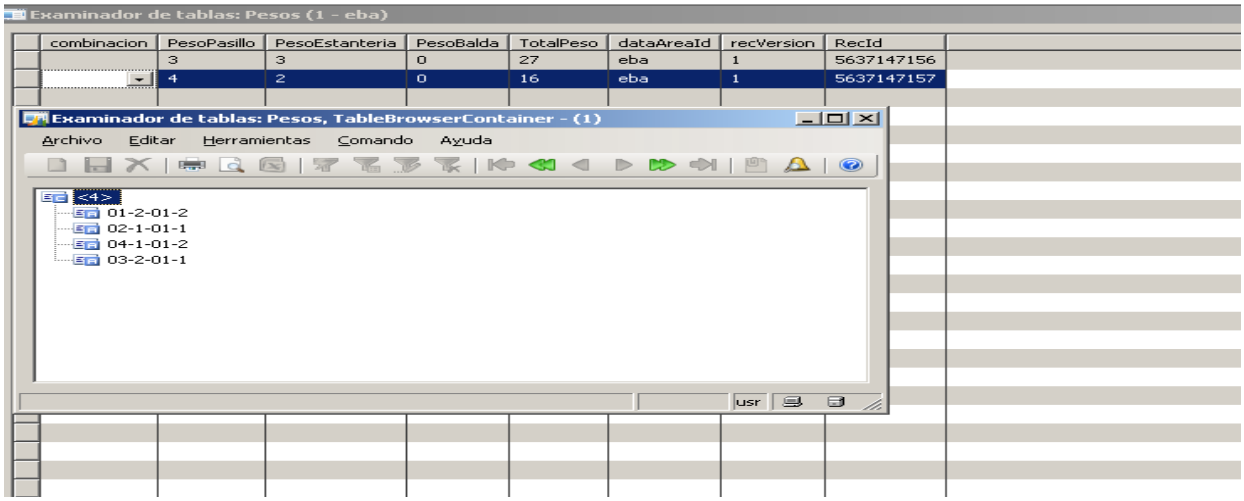
<4>
- 01-2-01-2
- 02-1-01-1
- 03-2-01-1
- 04-1-01-2

usr

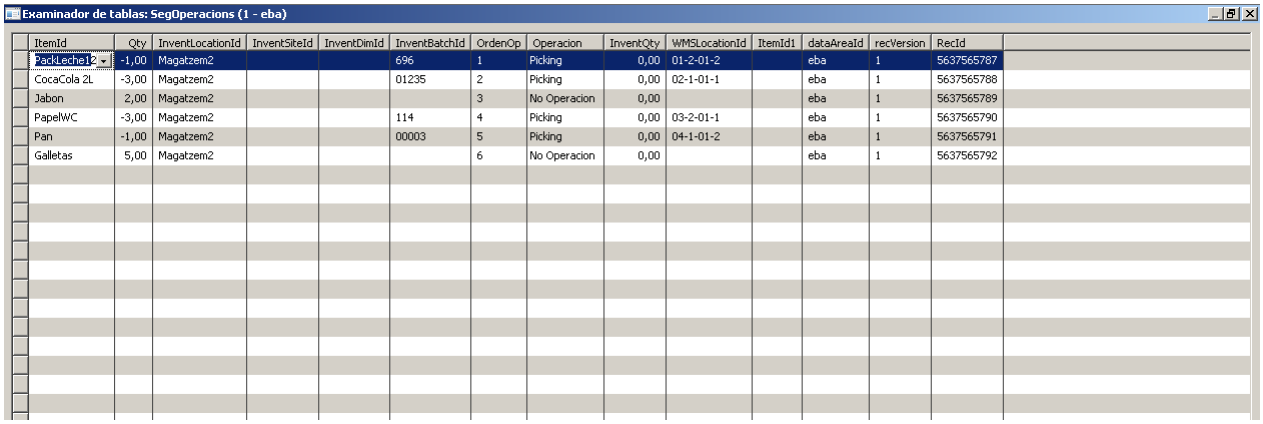
Il·lustració 99 Taula Permutar



Il·lustració 100 Taula Pesos



Il·lustració 101 Taula Pesos



Il·lustració 102 Taula SegOperacions

Examinador de tablas: TablaInforme (1 - eba)

ItemId	Qty	InventBatchId	InventLocationId	WMSLocationId	OrdenOp	dataAreaId	recVersion	RecId
PackLeche12	-1,00	696	Magatzem2	01-2-01-2	1	eba	1	5637145988
CocaCola 2L	-3,00	01235	Magatzem2	02-1-01-1	2	eba	1	5637145989
Pan	-1,00	00003	Magatzem2	04-1-01-2	3	eba	1	5637145990
PapelWC	-3,00	114	Magatzem2	03-2-01-1	4	eba	1	5637145991

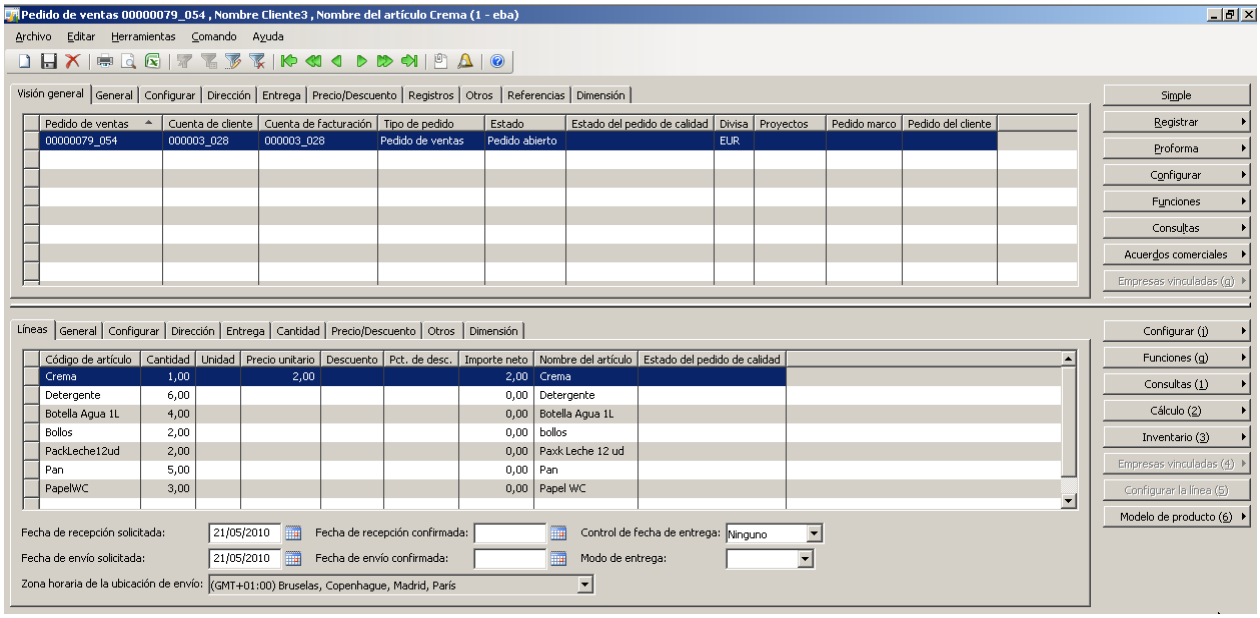
Il·lustració 103 Taula TablaInforme

L'informe que s'entregarà a l'usuari amb el resultat de la ruta de Picking és el següent:

Orden Operacion	Código de artículo	Número de lote	Cantidad	Ubicación	Almacén
1	PackLeche12ud	696	-1,00	01-2-01-2	Magatzem2
2	CocaCola 2L	01235	-3,00	02-1-01-1	Magatzem2
3	Pan	00003	-1,00	04-1-01-2	Magatzem2
4	PapelWC	114	-3,00	03-2-01-1	Magatzem2

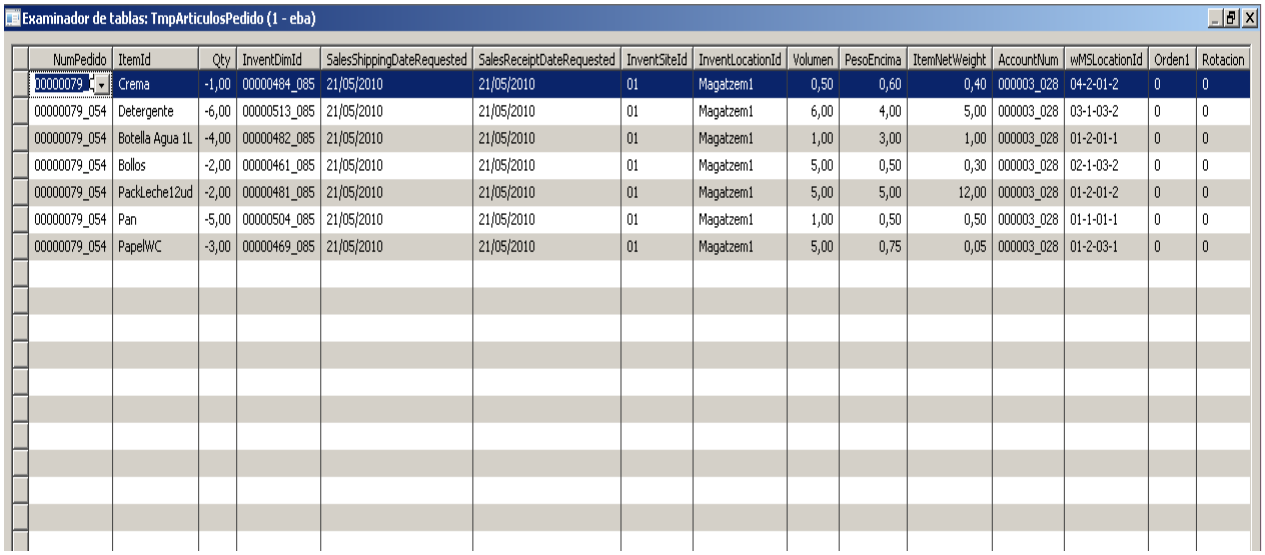
Il·lustració 104 Llistat de ruta de Picking PV 77_054

El quart exemple de comanda de venda és:



Il·lustració 105 Exemple comanda de venta

Els resultats obtinguts en les taules emprades en el procés per al primer magatzem són:



Examinador de tablas: TmpArticulosPedido (1 - eba)

NumPedido	ItemId	Qty	InventDimId	SalesShippingDateRequested	SalesReceiptDateRequested	InventSiteId	InventLocationId	Volumen	PesoEncima	ItemNetWeight	AccountNum	wMSLocationId	Orden1	Rotacion
00000079_	Crema	-1,00	00000484_085	21/05/2010	21/05/2010	01	Magatzem1	0,50	0,60	0,40	000003_028	04-2-01-2	0	0
00000079_054	Detergente	-6,00	00000513_085	21/05/2010	21/05/2010	01	Magatzem1	6,00	4,00	5,00	000003_028	03-1-03-2	0	0
00000079_054	Botella Agua 1L	-4,00	00000482_085	21/05/2010	21/05/2010	01	Magatzem1	1,00	3,00	1,00	000003_028	01-2-01-1	0	0
00000079_054	Bollos	-2,00	00000461_085	21/05/2010	21/05/2010	01	Magatzem1	5,00	0,50	0,30	000003_028	02-1-03-2	0	0
00000079_054	PackLeche12ud	-2,00	00000481_085	21/05/2010	21/05/2010	01	Magatzem1	5,00	5,00	12,00	000003_028	01-2-01-2	0	0
00000079_054	Pan	-5,00	00000504_085	21/05/2010	21/05/2010	01	Magatzem1	1,00	0,50	0,50	000003_028	01-1-01-1	0	0
00000079_054	PapelWC	-3,00	00000469_085	21/05/2010	21/05/2010	01	Magatzem1	5,00	0,75	0,05	000003_028	01-2-03-1	0	0

Il·lustració 106 Exemple TmpArticulosPedido

Examinador de tablas: TmpPedidosOrdenados (1 - eba)

Qty	InventBatchId	ItemId	InventLocationId	Ordenacion	WMSLocationId	InventDimId1	Dimension	Dimension	Dimension	SalesShippingDateRequested	SalesReceiptDateRequested	AccountNum	InventSiteId	dataAreaId
-2,00		Bollos	Magatzem1	6	02-1-03-2	00000461_085				21/05/2010	21/05/2010	000003_028	01	eba
-4,00		Botella Agua 1L	Magatzem1	3	01-2-01-1	00000482_085				21/05/2010	21/05/2010	000003_028	01	eba
-1,00		Crema	Magatzem1	5	04-2-01-2	00000484_085				21/05/2010	21/05/2010	000003_028	01	eba
-6,00		Detergente	Magatzem1	2	03-1-03-2	00000513_085				21/05/2010	21/05/2010	000003_028	01	eba
-2,00		PackLeche12ud	Magatzem1	1	01-2-01-2	00000481_085				21/05/2010	21/05/2010	000003_028	01	eba
-5,00		Pan	Magatzem1	7	01-1-01-1	00000504_085				21/05/2010	21/05/2010	000003_028	01	eba
-3,00		PapelWC	Magatzem1	4	01-2-03-1	00000469_085				21/05/2010	21/05/2010	000003_028	01	eba

Il·lustració 107 Exemple TmpPedidosOrdenados

Examinador de tablas: TmpCaducidad (1 - eba)

ItemId	InventBatchId	expDate	SalesShippingDateRequested	SalesReceiptDateRequested	Dimension	Dimension	Dimension	Valoracion	Rotacion	dataAreaId	recVersion	RecId
PackLeche1	696	30/06/2010	21/05/2010	21/05/2010				FIFO	40	eba	1	5637153097
Detergente	36369	30/06/2010	21/05/2010	21/05/2010				FIFO	40	eba	1	5637153098
Botella Agua 1L	00007	30/06/2010	21/05/2010	21/05/2010				FIFO	40	eba	1	5637153099
PapelWC	114	30/06/2010	21/05/2010	21/05/2010				FIFO	40	eba	1	5637153100
Crema	2325	30/06/2010	21/05/2010	21/05/2010				FIFO	40	eba	1	5637153101
Bollos	11114	30/06/2010	21/05/2010	21/05/2010				FIFO	40	eba	1	5637153102
Pan	00003	30/06/2010	21/05/2010	21/05/2010				FIFO	40	eba	1	5637153103

Il·lustració 108 Exemple TmpCaducidad

Examinador de tablas: CrearPicking (1 - eba)

ItemId	InventDimId	WMSLocationId	InventBatchId	Qty	InventLocationId	InventSiteId	dataAreaId	recVersion	RecId
PackLeche12		01-2-01-2	696	-2,00	Magatzem1	01	eba	1	5637151847
Detergente	00000509_085	03-1-03-2	36369	-6,00	Magatzem1	01	eba	1	5637151848
Botella Agua 1L	00000265_085	01-2-01-1	00007	-4,00	Magatzem1	01	eba	1	5637151849
PapelWC		01-2-03-1	114	-3,00	Magatzem1	01	eba	1	5637151850
Crema		04-2-01-2	2325	-1,00	Magatzem1	01	eba	1	5637151851
Bollos	00000489_085	02-1-03-2	11114	-2,00	Magatzem1	01	eba	1	5637151852
Pan		01-1-01-1	00003	-5,00	Magatzem1	01	eba	1	5637151853

Il·lustració 109 Exemple CrearPicking

Examinador de tablas: EstocDisponible (1 - eba)

ItemId	InventBatchId	Qty	dataAreaId	recVersion	RecId
PackLeche12	696	58,00	eba	1	5637151483
Botella Agua 1L	00007	16,00	eba	1	5637151484
PapelWC	114	197,00	eba	1	5637151485
Bollos	11114	8,00	eba	1	5637151486
Pan	00003	109,00	eba	1	5637151487

Il·lustració 110 Exemple EstocDisponible

Examinador de tablas: Permutar (1 - eba)

Combinacion	Pesos	dataAreaId	recVersion	RecId
	0	eba	1	5637926043
	0	eba	1	5637926067

Examinador de tablas: Permutar, TableBrowserContainer - (1)

Archivo Editar Herramientas Comando Ayuda

<5>
01-2-01-2
01-2-01-1
01-1-01-1
01-2-03-1
02-1-03-2

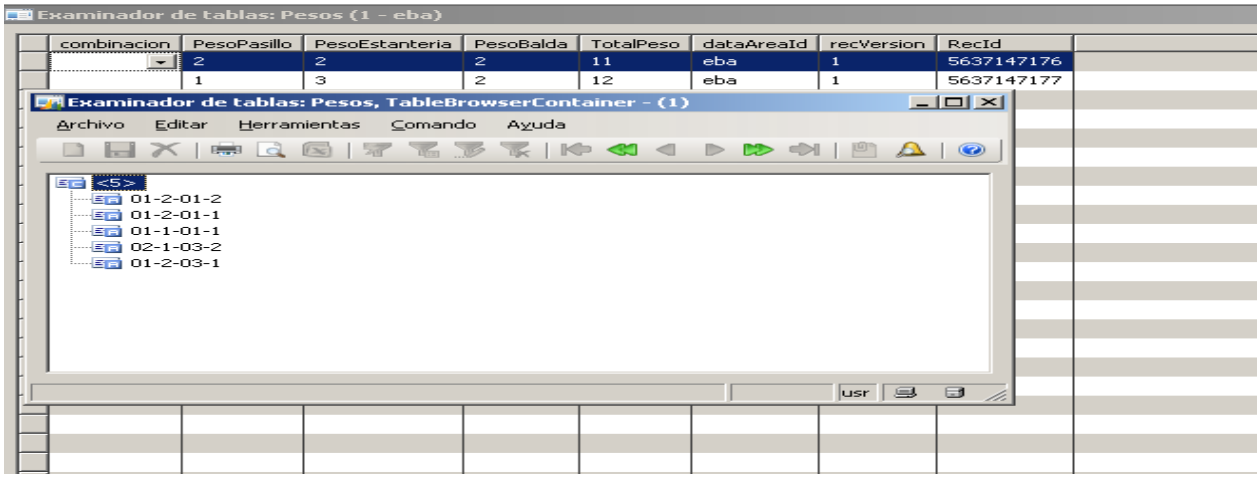
Examinador de tablas: Permutar, TableBrowserContainer - (1)

Archivo Editar Herramientas Comando Ayuda

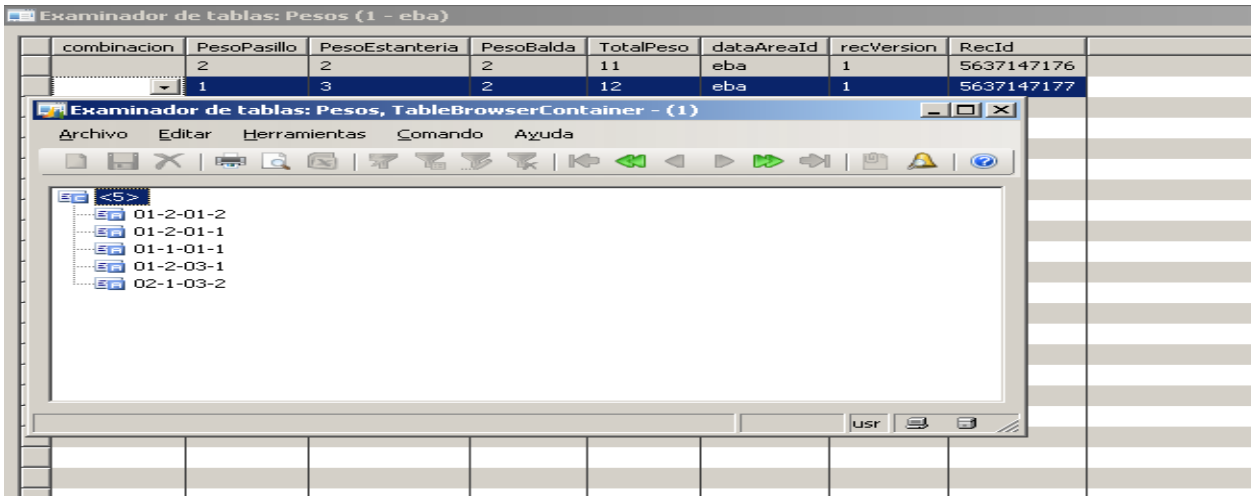
<6>
01-2-01-2
01-2-01-1
01-1-01-1
02-1-03-2
01-2-03-1

usr

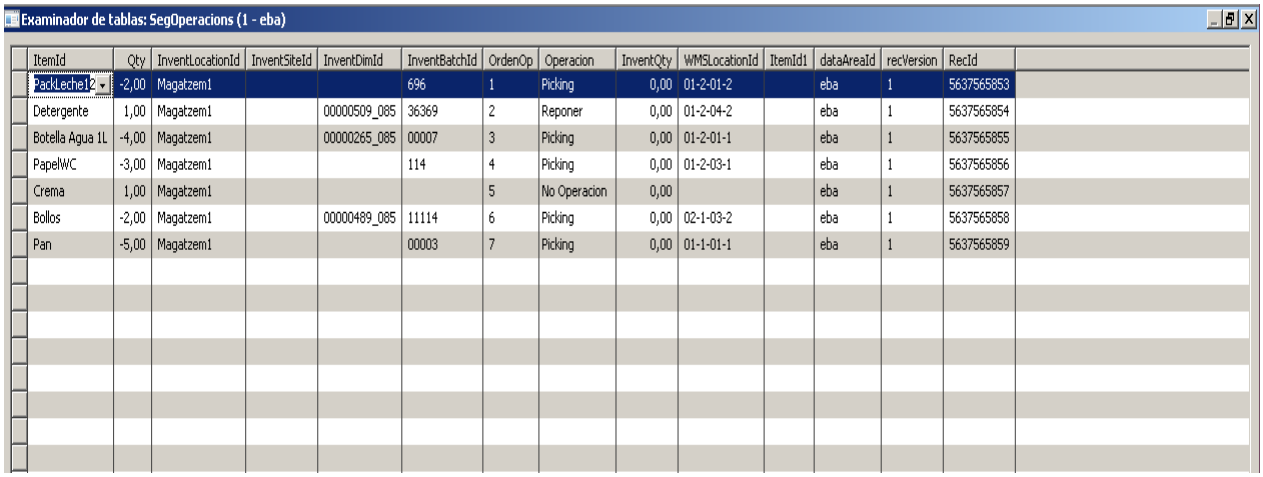
Il·lustració 111 Exemple Permutar



Il·lustració 112 Exemple Pesos



Il·lustració 113 Exemple Pesos



ItemId	Qty	InventLocationId	InventSiteId	InventDimId	InventBatchId	OrdenOp	Operacion	InventQty	WMSLocationId	ItemId1	dataAreaId	recVersion	RecId
PackLeche12	-2,00	Magatzem1			696	1	Picking	0,00	01-2-01-2		eba	1	5637565853
Detergente	1,00	Magatzem1		00000509_085	36369	2	Reponer	0,00	01-2-04-2		eba	1	5637565854
Botella Agua 1L	-4,00	Magatzem1		00000265_085	00007	3	Picking	0,00	01-2-01-1		eba	1	5637565855
PapelWC	-3,00	Magatzem1			114	4	Picking	0,00	01-2-03-1		eba	1	5637565856
Crema	1,00	Magatzem1				5	No Operacion	0,00			eba	1	5637565857
Bollos	-2,00	Magatzem1		00000489_085	11114	6	Picking	0,00	02-1-03-2		eba	1	5637565858
Pan	-5,00	Magatzem1			00003	7	Picking	0,00	01-1-01-1		eba	1	5637565859

Il·lustració 114 Exemple SegOpreacions

Examinador de tablas: TablaInforme (1 - eba)

ItemId	Qty	InventBatchId	InventLocationId	WMSLocationId	OrdenOp	dataAreaId	recVersion	RecId
PackLeche12	-2,00	696	Magatzem1	01-2-01-2	1	eba	1	5637146023
Botella Agua 1L	-4,00	00007	Magatzem1	01-2-01-1	2	eba	1	5637146024
Pan	-5,00	00003	Magatzem1	01-1-01-1	3	eba	1	5637146025
Bollos	-2,00	11114	Magatzem1	02-1-03-2	4	eba	1	5637146026
PapelWC	-3,00	114	Magatzem1	01-2-03-1	5	eba	1	5637146027

Il·lustració 115 Exemple TablaInforme

L'informe que s'entregarà a l'usuari amb el resultat de la ruta de Picking és el següent:

Orden Operacion	Código de artículo	Número de lote	Cantidad	Ubicación	Almacén
1	PackLeche12ud	696	-2,00	01-2-01-2	Magatzem1
2	Botella Agua 1L	00007	-4,00	01-2-01-1	Magatzem1
3	Pan	00003	-5,00	01-1-01-1	Magatzem1
4	Bollos	11114	-2,00	02-1-03-2	Magatzem1
5	PapelWC	114	-3,00	01-2-03-1	Magatzem1

Il·lustració 116 Llistat de ruta de Picking PV 79_054

Es realitza la mateixa comanda de venta que l'anterior però aquets ara es servirà des del segon magatzem.

Els resultats obtinguts en les taules emprades en el procés són:

Examinador de tablas: TmpArticulosPedido (1 - eba)

NumPedido	ItemId	Qty	InventDimId	SalesShippingDateRequested	SalesReceiptDateRequested	InventSiteId	InventLocationId	Volumen	PesoEncima	ItemNetWeight	AccountNum	wMSLocationId	Orden1	Rotacion
00000080_	Crema	-1,00	00000483_085	21/05/2010	21/05/2010	01	Magatzem2	0,50	0,60	0,40	000003_028	03-1-03-1	0	0
00000080_054	Detergente	-6,00	00000514_085	21/05/2010	21/05/2010	01	Magatzem2	6,00	4,00	5,00	000003_028	04-2-03-1	0	0
00000080_054	Botella Agua 1L	-4,00	00000485_085	21/05/2010	21/05/2010	01	Magatzem2	1,00	3,00	1,00	000003_028	01-2-01-1	0	0
00000080_054	Bollos	-2,00	00000441_085	21/05/2010	21/05/2010	01	Magatzem2	5,00	0,50	0,30	000003_028	04-2-01-1	0	0
00000080_054	PackLeche12ud	-2,00	00000505_085	21/05/2010	21/05/2010	01	Magatzem2	5,00	5,00	12,00	000003_028	01-2-01-2	0	0
00000080_054	Pan	-5,00	00000508_085	21/05/2010	21/05/2010	01	Magatzem2	1,00	0,50	0,50	000003_028	04-1-01-2	0	0
00000080_054	PapelWC	-3,00	00000498_085	21/05/2010	21/05/2010	01	Magatzem2	5,00	0,75	0,05	000003_028	03-2-01-1	0	0

Il·lustració 117 Exemple TmpArticulosPedido

Examinador de tablas: TmpPedidosOrdenados (1 - eba)

Qty	InventBatchId	ItemId	InventLocationId	Ordenacion	WMSLocationId	InventDimId1	Dimension	Dimension	Dimension	SalesShippingDateRequested	SalesReceiptDateRequested	AccountNum	InventSiteId	dataAreaId
-2,00		Bollos	Magatzem2	6	04-2-01-1	00000441_085				21/05/2010	21/05/2010	000003_028	01	eba
-4,00		Botella Agua 1L	Magatzem2	3	01-2-01-1	00000485_085				21/05/2010	21/05/2010	000003_028	01	eba
-1,00		Crema	Magatzem2	5	03-1-03-1	00000483_085				21/05/2010	21/05/2010	000003_028	01	eba
-6,00		Detergente	Magatzem2	2	04-2-03-1	00000514_085				21/05/2010	21/05/2010	000003_028	01	eba
-2,00		PackLeche12ud	Magatzem2	1	01-2-01-2	00000505_085				21/05/2010	21/05/2010	000003_028	01	eba
-5,00		Pan	Magatzem2	7	04-1-01-2	00000508_085				21/05/2010	21/05/2010	000003_028	01	eba
-3,00		PapelWC	Magatzem2	4	03-2-01-1	00000498_085				21/05/2010	21/05/2010	000003_028	01	eba

Il·lustració 118 Exemple TmpPedidosOrdenados

Examinador de tablas: TmpCaducidad (1 - eba)

ItemId	InventBatchId	expDate	SalesShippingDateRequested	SalesReceiptDateRequested	Dimension	Dimension	Dimension	Valoracion	Rotacion	dataAreaId	recVersion	RecId
PackLeche12	696	30/06/2010	21/05/2010	21/05/2010				FIFO	40	eba	1	5637153104
Detergente	36369	30/06/2010	21/05/2010	21/05/2010				FIFO	40	eba	1	5637153105
Botella Agua 1L	00007	30/06/2010	21/05/2010	21/05/2010				FIFO	40	eba	1	5637153106
PapelWC	114	30/06/2010	21/05/2010	21/05/2010				FIFO	40	eba	1	5637153107
Crema	2325	30/06/2010	21/05/2010	21/05/2010				FIFO	40	eba	1	5637153108
Bollos	11114	30/06/2010	21/05/2010	21/05/2010				FIFO	40	eba	1	5637153109
Pan	00003	30/06/2010	21/05/2010	21/05/2010				FIFO	40	eba	1	5637153110

Il·lustració 119 Exemple TmpCaducidad

Examinador de tablas: CrearPicking (1 - eba)

ItemId	InventDimId	WMSLocationId	InventBatchId	Qty	InventLocationId	InventSiteId	dataAreaId	recVersion	RecId
PackLeche12		01-2-01-2	696	-2,00	Magatzem2	01	eba	1	5637151854
Detergente	00000511_085	04-2-03-1	36369	-6,00	Magatzem2	01	eba	1	5637151855
Botella Agua 1L		01-2-01-1	00007	-4,00	Magatzem2	01	eba	1	5637151856
PapelWC		03-2-01-1	114	-3,00	Magatzem2	01	eba	1	5637151857
Crema		03-1-03-1	2325	-1,00	Magatzem2	01	eba	1	5637151858
Bollos	00000490_085	04-2-01-1	11114	-2,00	Magatzem2	01	eba	1	5637151859
Pan		04-1-01-2	00003	-5,00	Magatzem2	01	eba	1	5637151860

Il·lustració 120 Exemple CrearPicking

Examinador de tablas: EstocDisponible (1 - eba)

ItemId	InventBatchId	Qty	dataAreaId	recVersion	RecId
PackLeche12	696	58,00	eba	1	5637151488
Botella Agua 1L	00007	65,00	eba	1	5637151489
PapelWC	114	197,00	eba	1	5637151490
Bollos	11114	8,00	eba	1	5637151491
Pan	00003	109,00	eba	1	5637151492

Il·lustració 121 Exemple EstocDisponible

Examinador de tablas: Permutar (1 - eba)

Combinacion	Pesos	dataAreaId	recVersion	RecId
	0	eba	1	5637926163
	0	eba	1	5637926187

Examinador de tablas: Permutar, TableBrowserContainer - (1)

Archivo Editar Herramientas Comando Ayuda

<5>
01-2-01-2
01-2-01-1
04-1-01-2
03-2-01-1
04-2-01-1

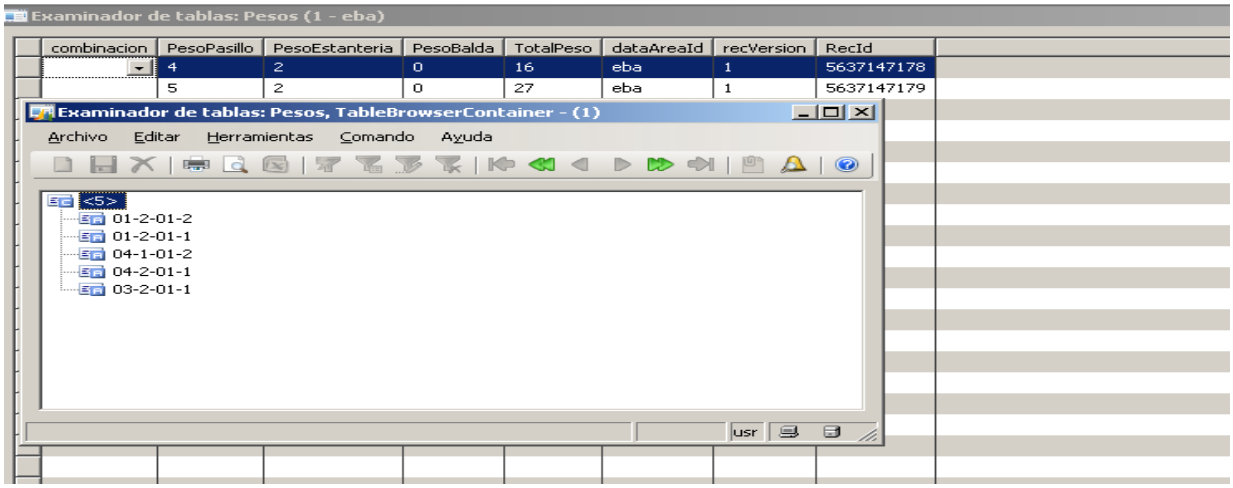
Examinador de tablas: Permutar, TableBrowserContainer - (1)

Archivo Editar Herramientas Comando Ayuda

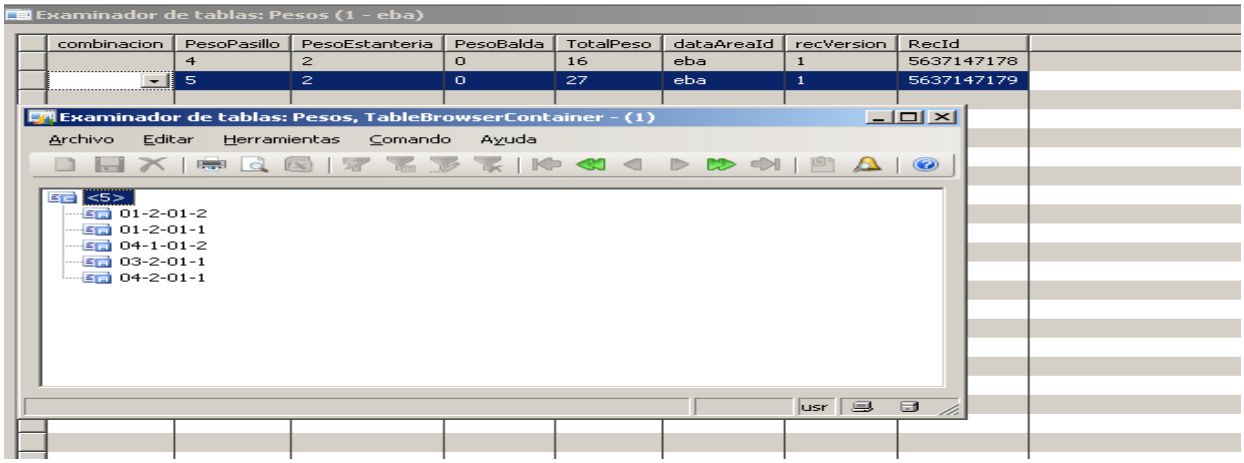
<5>
01-2-01-2
01-2-01-1
04-1-01-2
04-2-01-1
03-2-01-1

usr

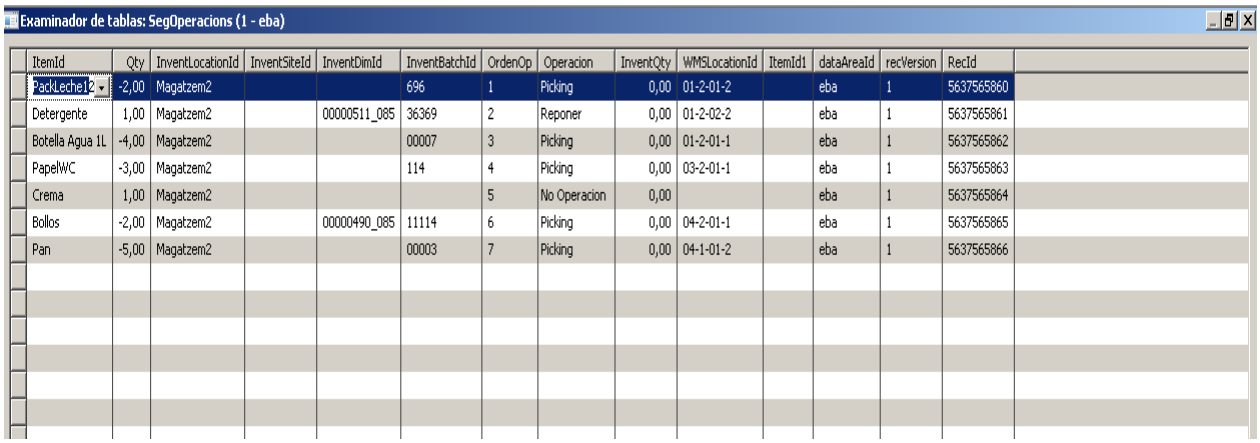
Il·lustració 122 Exemple Permutar



Il·lustració 123 Exemple Pesos



Il·lustració 124 Exemple Pesos



ItemId	Qty	InventLocationId	InventSiteId	InventDimId	InventBatchId	OrdenOp	Operacion	InventQty	WMSLocationId	ItemId1	dataAreaId	recVersion	RecId
PackLeche12	-2,00	Magatzem2			696	1	Picking	0,00	01-2-01-2		eba	1	5637565860
Detergente	1,00	Magatzem2		00000511_085	36369	2	Reponer	0,00	01-2-02-2		eba	1	5637565861
Botella Agua 1L	-4,00	Magatzem2			00007	3	Picking	0,00	01-2-01-1		eba	1	5637565862
PapelWC	-3,00	Magatzem2			114	4	Picking	0,00	03-2-01-1		eba	1	5637565863
Crema	1,00	Magatzem2				5	No Operacion	0,00			eba	1	5637565864
Bollos	-2,00	Magatzem2		00000490_085	11114	6	Picking	0,00	04-2-01-1		eba	1	5637565865
Pan	-5,00	Magatzem2			00003	7	Picking	0,00	04-1-01-2		eba	1	5637565866

Il·lustració 125 Exemple SegOperacions

Examinador de tablas: TablaInforme (1 - eba)

ItemId	Qty	InventBatchId	InventLocationId	WMSLocationId	OrdenOp	dataAreaId	recVersion	RecId
PackLeche12	-2,00	696	Magatzem2	01-2-01-2	1	eba	1	5637146028
Botella Agua 1L	-4,00	00007	Magatzem2	01-2-01-1	2	eba	1	5637146029
Pan	-5,00	00003	Magatzem2	04-1-01-2	3	eba	1	5637146030
Bollos	-2,00	11114	Magatzem2	04-2-01-1	4	eba	1	5637146031
PapelWC	-3,00	114	Magatzem2	03-2-01-1	5	eba	1	5637146032

Il·lustració 126 Exemple TablaInforme

L'informe que s'entregarà a l'usuari amb el resultat de la ruta de Picking és el següent:

Orden Operacion	Código de artículo	Número de lote	Cantidad	Ubicación	Almacén
1	PackLeche12ud	696	-2,00	01-2-01-2	Magatzem2
2	Botella Agua 1L	00007	-4,00	01-2-01-1	Magatzem2
3	Pan	00003	-5,00	04-1-01-2	Magatzem2
4	Bollos	11114	-2,00	04-2-01-1	Magatzem2
5	PapelWC	114	-3,00	03-2-01-1	Magatzem2

Il·lustració 127 Llistat de ruta de Picking PV 80_054